UCRL-ID-148221

The Stochastic Engine Initiative: Improving Prediction of Behavior in Geologic Environments We Cannot Directly Observe

R. Aines, J. Nitao, R. Newmark, S. Carle, A. Ramirez, D. Harris, J. Johnson, V. Johnson, D. Ermak, G. Sugiyama, W. Hanley, S. Sengupta, W. Daily, R. Glaser, K. Dyer, G. Fogg, Y. Zhang, Z. Yu, and R. Levine

U.S. Department of Energy

Lawrence
Livermore
National
Laboratory

May 9, 2002

DISCLAIMER

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

This work was performed under the auspices of the U. S. Department of Energy (DOE) by the University of California Lawrence Livermore National Laboratory (LLNL) under Contract W-7405-Eng-48. This research is funded by the Laboratory Directed Research and Development (LDRD) Program at LLNL. The LDRD Program is mandated by Congress to fund director-initiated, long-term research and development (R&D) projects in support of the DOE and national laboratories mission areas. The Director's Office LDRD Program at LLNL funds creative and innovative R&D to ensure the scientific vitality of the Laboratory in mission-related scientific disciplines.

The Stochastic Engine Initiative:

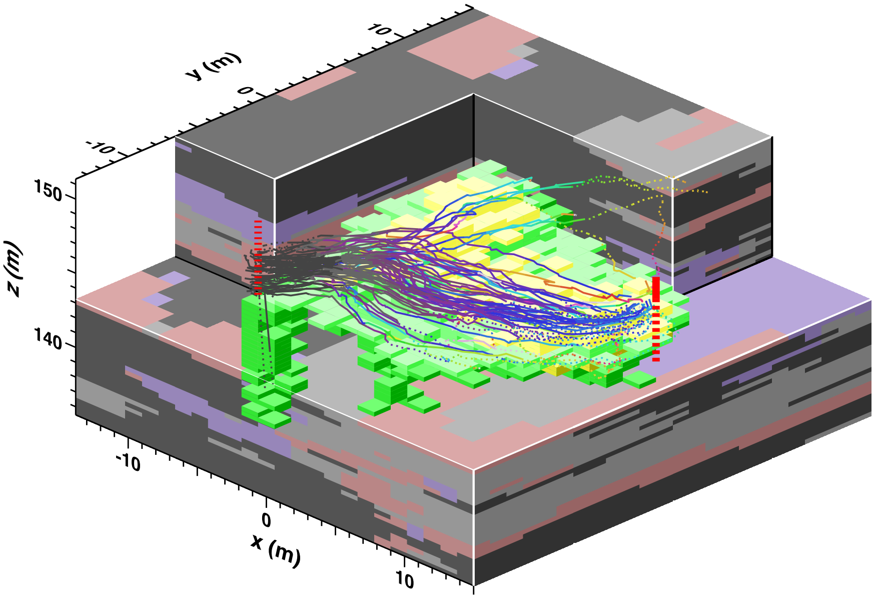
Improving Prediction of Behavior in Geologic Environments We Cannot Directly Observe

Roger Aines, John Nitao, Robin Newmark, Steve Carle, Abe Ramirez, Dave Harris, Jim Johnson, Virginia Johnson, Don Ermak, and Gayle Sugiyama
LLNL Energy and Environment

Bill Hanley, Sailes Sengupta, Bill Daily, and Ron Glaser
LLNL Engineering

Kathleen Dyer
LLNL Computations

Graham Fogg, Yong Zhang, Zhaoxia Yu, and Richard Levine
University of California at Davis



Contents

Abstract

The stochastic engine uses modern computational capabilities to combine simulations with observations. We integrate the general knowledge represented by models with specific knowledge represented by data, using Bayesian inferencing and a highly efficient staged Metropolis-type search algorithm. From this, we obtain a probability distribution characterizing the likely configurations of the system consistent with existing data. The primary use will be optimizing knowledge about the configuration of a system for which sufficient direct observations cannot be made. Programmatic applications include underground systems ranging from environmental contamination to military bunkers, optimization of complex non-linear systems, and timely decision-making for complex, hostile environments such as battlefields or the detection of secret facilities.

We create a stochastic “base representation” of system configurations (states) from which the values of measurable parameters can be calculated using forward simulators. Comparison of these predictions to actual measurements drives embedded Bayesian inferencing, updating the distributions of states in the base representation using the Metropolis method. Unlike inversion methods that generate a single best-case deterministic solution, this method produces all the likely solutions, weighted by their likelihoods. This flexible method is best applied to highly non-linear, multi-dimensional problems.

Staging of the Metropolis searches permits us to run the simplest model systems, such as lithology estimators, at the lower stages. The majority of possible configurations are thus eliminated from further consideration by more complex simulators, such as flow and transport models. Because the method is fully automated, large data sets of a variety of types can be used to refine the system configurations. The most important prerequisites for optimal use of this method are well-characterized forward simulators, realistic base representations, and most importantly an ability to obtain disparate data sets that are directly affected by the system configuration. Our initial earth-sciences application uses models for lithology, flow and transport, geochemistry, and geophysical imaging; the system configuration (base representation) being refined is the rock type at each underground location.

In the initial stages of this initiative we demonstrated a two-stage analysis of synthetic Electrical Resistance Tomography (ERT) data and hydraulic flow information (Newmark et al., 2002). We used these results to develop algorithms that improve efficiency of the Metropolis search and provide accurate diagnostic evaluation during the search. Using actual data from a highly contaminated A/M outfall and solvent tank storage areas at the Savannah River Site (SRS), we used the stochastic engine to resolve lithology using ERT data. SRS will use these methods in their design and implementation of steam cleanup of the largest trichloroethylene (TCE) source in the Department of Energy (DOE) complex. We have implemented “soft conditioning” algorithms that allow us to use a variety of data types to control the initial representations, and most importantly, to use the final distribution resulting from one stochastic engine analysis as the initial distribution for a subsequent analysis. We have created a web-based interface that will allow collaborators like SRS to enter data and observe results of calculations on Lawrence Livermore National Laboratory (LLNL) supercomputers in an interactive mode. All engine functions operate in three dimensions, and a parallel implementation on Linux cluster machines is in initial testing. The method will be extended to include active process analysis, in which an ongoing data stream is used to continuously update the understanding of the system configuration. Applications to other types of state spaces, such as chemical parameters in a reacting system or atmospheric plume movement, are being evaluated.

Introduction

The fundamental problem in earth sciences is to determine the properties of an object that we cannot directly observe. This problem is common to numerous other areas of investigation. We use inference and models to extrapolate or interpolate our knowledge, but we are fundamentally limited by the inability to inquire over the entire spatial or temporal domain of interest. The purpose of the Stochastic Engine Initiative is to attack this problem by developing a method to simultaneously use many kinds of data to refine our understanding of complex geologic systems. We focus on improving one "base" set of data (or representation of the system), from which other parameters of interest can be calculated using process models. The lithology (the general physical characteristics of a rock) of an underground system is the base representation for geological systems. It provides a ready means to predict the behavior of the system under forcing events such as injection of a fluid; when we know the lithology more accurately, we can predict the behavior of the system more accurately. The response of the system to these forcing events can then be measured to further improve the knowledge of the system. This feedback is central to our ability to acquire enough knowledge about complicated systems; we need to utilize each layer of knowledge to improve our acquisition of new data, continuously improving the detail and accuracy of our system knowledge. The stochastic engine is designed to incorporate everything from the geologists' first field observations to the millions of measurements made during a field operation such as a steam remediation project, into an integrated and continuously improving understanding of the base representation. While our first application is geologic, this method is broadly applicable to any topical area in which direct observations of a system can be combined with general understanding represented by simulation.

Technology of this kind is needed by all large-scale subsurface efforts. The most obvious, such as oil recovery, are those in which the cost of additional wells is very large and the goal is to maximize recovery per dollar invested. In government applications such as nuclear waste disposal or environmental remediation, a more immediate goal is often the reduction of uncertainty in the outcome of costly or long-term efforts. These data-rich applications are in contrast to data-poor situations, such as locating underground structures. In all cases the stochastic engine can improve the value of existing data, and guide the acquisition of future data through quantitative evaluation of method, location, and number of points. Because geologic investigations proceed on a time scale of days to years, the stochastic engine can serve as a real-time analysis tool, updating the global understanding of the system parameters as each new item is acquired. We are working with partners in the Department of Energy (Savannah River Site), the Department of Defense (Navy—ESTCP program) and the Environmental Protection Agency to apply the stochastic engine to major environmental cleanup efforts. These data-rich systems are desperately in need of a method to unify all data types, and provide real measures of uncertainty to guide decisions.

How can we understand a system that is too complex to sample or impossible to observe directly, but for which we have good models that will predict behavior under specific conditions?

Such systems are a large part of our LLNL mission. They include not only existing underground systems, but also problems in the future such as nuclear waste disposal systems; complex or hostile environments on battle fields or in secret facilities; and problems in which short response times preclude fielding additional equipment to more fully characterize the situation. The stochastic engine addresses these problems by integrating the general knowledge represented by models with specific knowledge represented by data. Its development constitutes a technological leap in the integration of simulation and data.

The stochastic engine honors all data and model information to produce probability distributions identifying likely system configurations or behavior, and quantifying the potential improvement provided by new data. Even when conventional inversion and analysis methods are able to address complex problems, they provide only a single "best" answer, throwing away much of the information and precluding other likely possibilities. This hinders the subsequent use of the analysis by failing to allow for alternative possible outcomes.

For instance, waste package engineers for the Yucca Mountain Program (YMP) do not want to know a single water chemistry that can contact a package, but rather: the range of chemistries; the likelihoods of those chemistries; and where they are expected to occur. The stochastic engine will answer this kind of question. The optimal application will be in projects where good simulators exist but data are incomplete or complex. This includes many earth science problems as well as intelligence gathering and tomographic imaging. Most importantly, the stochastic engine allows continuous integration of new data into the analysis, improving understanding and reducing uncertainty in the areas where it is most valuable. The stochastic engine is intended to be an integral part of long-term programs. This is a radically new approach to understanding and predicting complex systems.

Approach

The stochastic engine uses existing simulators to predict data values that are then compared to exactly analogous measurements to determine which possible configurations of a system are in fact closest to the real condition. An extremely efficient search algorithm, derived from the Metropolis/Hastings method, is used to determine which states to test. Different types of data (and their accompanying simulation) can be combined in stages, so that extremely complex state spaces can be searched quickly. The initial state space is described by a mathematical model called the base representation that includes all the salient features of the system, while being as simple as possible. This constitutes the "prior" distribution in the Bayesian inferencing scheme. The results of the engine analysis are in terms of these same states; the "posterior" distribution resulting from the analysis is the set of states that are consistent with the data and inherent error in the system.

The simulators used are all forward models, that is, they predict a value given an initial condition; these can be used with extremely non-linear problems which are difficult or impossible to directly invert. Modern computational power makes this reasonable for complicated problems, but the search algorithm has proven to be so efficient that many problems of geologic interest are tractable on workstations alone. The wide range of applicable problems is a function of the number of good models (simulators) that exist today. The stochastic engine methodology can use any model that predicts results based on initial conditions. Initial development focuses on earth-sciences models for lithology, flow and transport, geochemistry, and geophysical imaging.

Contributing data can be of many types, ranging from distinct physical or chemical measurements for which sensitivity, resolution, etc., are known, to “soft” data such as expert inference or qualitative models. The experimental methods used include multiple simultaneous imaging methods and “active” analyses such as pump tests or deformation tests that force changes in the system that are predictable if the internal structure is known. The resulting analysis is unique both in the simultaneous use of multiple data types (for instance x-ray tomography and positron emission tomography) and in the calculation of the structure directly in terms of probabilities. Rather than a single “best” structure, the stochastic engine generates a range of plausible structures and the corresponding probabilities that they are correct. This facilitates decision analysis and needs-based experimental planning.

We are focusing on geological/ geophysical systems for which extensive observations can be made on time scales shorter than the characteristic scale for the problem, enabling predictive understanding to evolve much faster than the real-time evolution of the natural system. These include remediation of groundwater, atmospheric transport, and characterization of natural environments. We have good process models and statistical means of describing initial conditions in systems involved in environmental management, nuclear waste, and carbon management, and in other natural systems where complexity hinders prediction of future outcomes.

The stochastic engine has been tested on two sets of real data from the Savannah River Site (SRS). All engine functions operate in three dimensions, and testing on cluster machines has begun. A web-based user interface allows remote users (such as our collaborators at SRS) to run the engine on our LLNL machines, with full interactivity.

The application of the stochastic engine for national security issues involving large amounts of uncertain data was an obvious extension, and the events of September 11 2001 encouraged us to accelerate that phase of development. Applications in intelligence and defense areas include sampling strategies and evaluation of facilities involved in weapons of mass destruction, imaging and non-destructive evaluation of complex assemblies, measuring structural response to deformation forces, and locating the source of atmospheric plumes.

Work in Progress

This report describes the state of development of the Stochastic Engine Initiative at the halfway point of an expected three-year effort. An initial description of the project was given by Newmark et al. (2002), including future development efforts that are part of our current plan.

The Stochastic Engine

Overview

One of the fundamental theorems of conditional probability is known as Bayes theorem. The result relates the probability of one event given the occurrence of another (e.g.; A given B occurred), to the inverse conditional probability (e.g.; B given A has occurred) as follows:

$$P(B|A) = \frac{P(B)P(A|B)}{P(A)}$$

The probability of B given A has occurred, denoted $P(B|A)$, is called the posterior probability, while $P(B)$ alone is the prior probability of event B. One manifestation of this theorem has B as a hypothesis being tested and A as an observation pertinent to that hypothesis. Hence, Bayes theorem allows one to revise the initial probability of a hypothesis ($P(B)$) by incorporating observed data (A) to produce an updated (and more accurate in light of the new data) probability of the hypothesis given the observed data ($P(B|A)$).

For the current approach, Bayes theorem will allow us to model our unknown parameters as random quantities with corresponding probability distributions defined on the space of possible parameter values. This representation of the unknown parameters as random quantities rather than fixed unknown values is a critical distinction between our approach and prior efforts in the earth sciences.

The actual connection of a hypothesis to an observation is made via a forward model: for a possible subsurface configuration the forward model predicts the values that would be observed by actual measurement. These are compared to real data. The degree of match between the real and predicted data is fed back to a Markov Chain Monte Carlo (MCMC) algorithm that samples candidate lithologic configurations for testing. Accepted states constitute samples from the posterior distribution and provide the basis for subsequent inference. By staging these comparisons in a series of MCMC algorithms, we can identify probable configurations using fast models early in the process. The most computationally intensive models are only used at later stages on configurations that are already known to be consistent with data used at earlier stages (Figure 1).

Configurations that pass all the stages are possible true configurations of the system. It is usually the case that the MCMC approach produces a reduction in the number of possible configurations (represented in the prior distribution) by many orders of magnitude. The approach also provides a seamless methodology for combining observational data with our forward models to produce state estimates and corresponding uncertainties that are not readily available through conventional inversion approaches. This is an extremely powerful method for incorporating previously known information and newly acquired observations into an estimate of the probability distribution of the states of the system. By generating likelihoods of actual lithologies, we can readily involve a variety of data types in the inference process and use the obtained lithologic posterior distribution to guide further investigations. By combining configurations into meta-classes (configurations that are so similar as to behave identically in the field, within error) we can readily deduce whether there is more than one highly probable

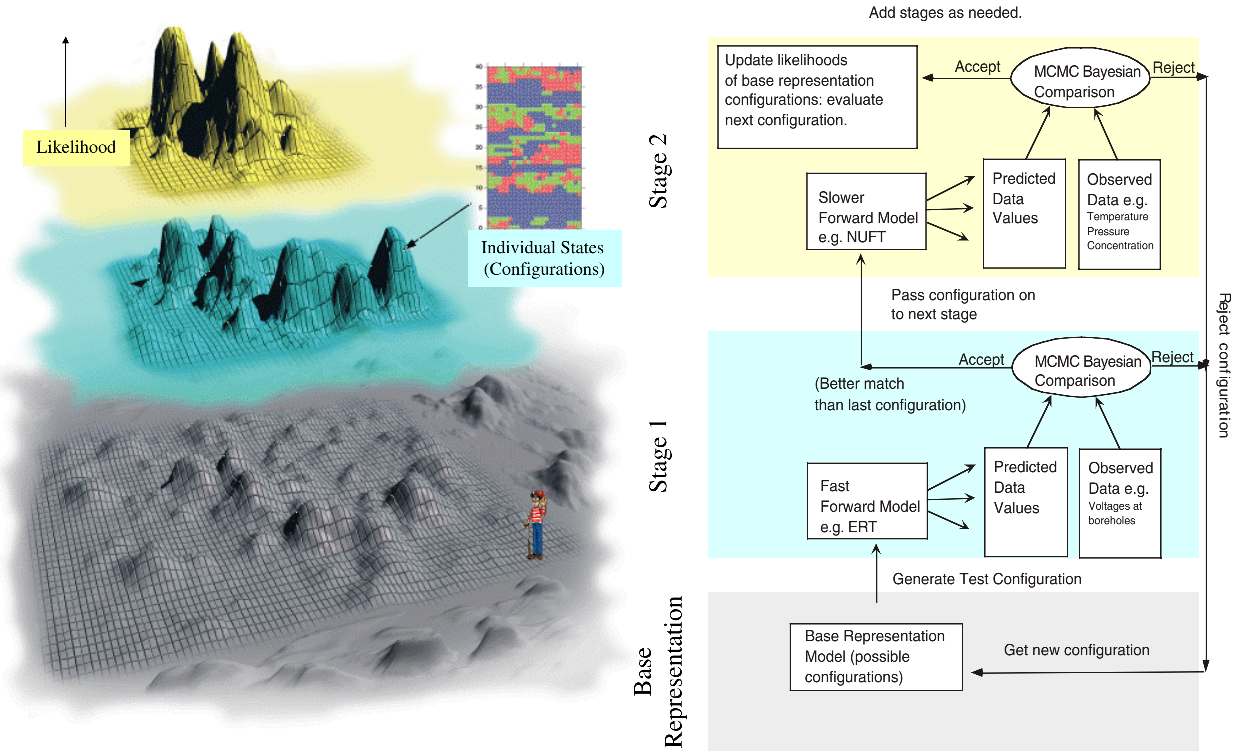


Figure 1. The MCMC Stochastic Engine combines observations and simulations to determine the likelihoods of possible system configurations.

configuration for our system, and which data will be the most useful in resolving between competing configurations.

The MCMC staged algorithm is well suited to a number of improvements that we anticipate will be crucial to dealing with complex, three-dimensional problems:

- Any number of stages can be used, involving all the data available for the system.
- Initial constraints placed on the base model confine the analysis to solutions that are known to be physically realistic, speeding the search and enhancing the usefulness of the answer.
- Data can be added to the algorithm sequentially, as it becomes available.
- The algorithm can be stopped when all available information has been processed, and the newly obtained distribution of the possible configurations can then be the basis for processing new data in a subsequent staged MCMC algorithm.
- Resolution in the base model can vary across the state space, allowing focus on critical areas. Individual spatial volumes can be analyzed by their own stages, and the result collapsed to a single probability distribution (as described above).
- Additional parameters can be added to the analysis by mapping them onto the lithologic representation. For instance, the presence of contaminant can be added as a representation element and a series of stages for chemical data types can be incorporated into the simulation to resolve the location of the contaminant.

We believe that this approach represents a true breakthrough in dealing with the combinatorial avalanche that has limited true statistical analysis in the earth sciences. The ability to stage the MCMC analysis and rapidly winnow the searched states makes it possible for the stochastic engine to evaluate complex problems on large scales using complex models.

Our goals are feasible but challenging. In order to maximize our progress and the usability of the end product, it is important to deal with tractable applications. Those will be problems in which: we have familiarity with the lithologic properties (alluvial soil systems); a rich variety of disparate data exists (a thoroughly monitored steam injection system); chances to reiterate the analysis of the system occur (common in multiple injections conducted for creosote sites); and in which there is good engineering control of the system. These give us a well-poised problem with defensible priors and well-described data for the updating process. We anticipate that for a large site like the Savannah River steam injection site, the initial application phases would require cluster level computational resources. As the approach is refined and the underground system better understood, the forward computational aspects could be transferred to more commonly available resources such as those used to run NUFT simulations today.

Mathematical Basis

This tool is undergoing continuing revision and improvement. The major components and their functions are:

Markov Chain Monte Carlo—The Markov Chain Monte Carlo (MCMC) methodology provides a flexible framework that can be adapted to perform a variety of analyses and inference tasks. It uses a Markov chain state/transition structure to control the sampling process. MCMC techniques enable sampling from a posterior distribution by representing it as the stationary distribution of a Markov chain, which is simulated until the chain achieves equilibrium. At that point the chain is generating a sequence of samples from the posterior (i.e., target) distribution. For Bayesian analysis, we are able to adopt the approach to simulate and estimate posterior distributions that embody our available prior information (e.g., historical data and phenomenology models) and newly acquired observational data. MCMC algorithms can assume a variety of forms with the most useful to us being the Metropolis framework.

Metropolis Algorithm—The basic stochastic engine approach is a derivative of the Metropolis algorithm (Metropolis et. al., 1953) described by Mosegaard and Tarantola (1995). This particular MCMC algorithm has demonstrated potential in solving inverse problems involving complex physical systems and supports several key enhancements necessary to mitigate the combinatorial demands underlying the MCMC methodology. For this framework, the solution to an inverse problem is an estimate of the posterior probability distribution defined on the corresponding space S of possible solutions. In other words, for any potential solution $s_0 \in S$, the stochastic engine will provide an estimate of the probability and confidence that s_0 is indeed the true solution to the given system. This allows future analysis to focus upon the most likely explanations of system behavior—thereby improving both the efficiency and effectiveness of follow-on efforts. Moreover, since the Mosegaard and Tarantola's formulation directly implements Bayesian analysis, results generated by the stochastic engine (i.e., the estimated posterior distribution, predictions, hypothesis testing, model comparison, etc.) may be incrementally updated as more data become available over time.

The inverse problem under consideration may be described as follows. Let D and M denote the data space and model space respectively, and suppose that there exists a mapping G such that:

$$\underline{d} = G(\underline{m})$$

where $\underline{m} \in M$ is a parameter vector describing the system of interest and $\underline{d} \in D$ is a vector of measurements taken on that system. The inverse problem occurs when a vector of data values is observed, say $\underline{d}_0$, and we want to determine the value of the parameter vector $\underline{m}_0$ that gave rise to $\underline{d}_0$. Usually this problem is so poorly constrained (*i.e.* under determined) and highly nonlinear that the specification of a deterministic solution for $\underline{m}_0$ that is unique and possesses a high degree of confidence is virtually impossible. In these types of situations, a probabilistic solution to the inverse problem is generally superior to any classical deterministic optimization approach.

The Mosegaard and Tarantola version of the Metropolis algorithm produces a sequence of samples from the space of possible solutions M where the samples are generated at rates proportional to their posterior probabilities. The models generated most frequently are consistent with both our prior information on M and the observations being processed. In the long run, the posterior distribution can be estimated from the generated sample frequencies. Since the information used to drive the simulation is taken from two distinct sources (prior information and observational data) the sampling process can be viewed as consisting of two distinct components. The first component generates samples according to the a-priori distribution $\rho(m)$ on model space (these samples are called "proposal" samples and are possible solutions to the inverse problem). In the algorithm, this sampling process is manifested as a random walk through the state space M. The states of the random walk are the members of M and the one-step transition probabilities are designed to produce a long-run stationary distribution equal to $\rho(m)$. The second component takes the form of a decision process that either accepts or rejects the proposal sample generated from the a-priori random walk. This decision is based upon the likelihood that the proposed solution could have produced the observed data. Specifically, suppose that the current state of the random walk is $\underline{m}_i$ and that a randomized rule based upon the one-step transition probabilities propose a move to state $\underline{m}_j$. If these transitions were always accepted, then the simulation would be sampling from the prior distribution. But, instead suppose that the proposal transition is only accepted according to the following rules:

- 1) For both the current and proposal states $\underline{m}_i$ and $\underline{m}_j$, compute the respective likelihoods $L(\underline{m}_i)$ and $L(\underline{m}_j)$ that these models produced the observed data. (Note: These likelihood functions essentially measure the degree of fit between the observed data and the corresponding data predicted by the model.)
- 2) If $L(\underline{m}_j) \geq L(\underline{m}_i)$, then accept the proposed transition with probability 1 and move the random walk to state $\underline{m}_j$. (Note: The algorithm always accepts the transition when the new state provides a better explanation of the data than the current state.)
- 3) If $L(\underline{m}_j) < L(\underline{m}_i)$, then use a randomized decision rule and accept the proposed transition with probability $L(\underline{m}_j)/L(\underline{m}_i)$ and move the random walk to state $\underline{m}_j$.

Otherwise, transition back to state $\underline{m}_i$. (Note: By allowing the random walk to transition to a less likely state, the process can move out of a local extrema.)

We have demonstrated that the samples generated through this three-step process will have a limiting distribution that is proportional to the desired posterior distribution $\rho\left(\underline{m}|\underline{d}_0\right)$.

There are a variety of issues that must be addressed during the implementation of this methodology. The most fundamental concern is that the proposal random walk must be designed so that a limiting stationary distribution actually exists and the overall process converges. For this to happen, the transition probabilities must be defined so that the process is ergodic—in other words, it must be aperiodic and irreducible. Once we are guaranteed the simulation converges, the critical issue becomes convergence rate. Key factors affecting the convergence rate include: (i) the strength and quality of the prior distribution $\rho(m)$, (ii) the representation, resolution and dimensionality of the state space M, and (iii) the computation of the likelihood function for any given state of interest.

In general, the prior's impact on convergence originates from its control over which of the neighboring states will be proposed as the next state for the process to occupy. Once a proposal state is selected, the sensitivity of the likelihood function influences the proposal acceptance rate and in turn the overall convergence rate. Hence, both the prior distribution and likelihood effect how well the process mixes (i.e. samples) the support of the posterior distribution. This is arguably the most critical function in the MCMC paradigm since the more rapidly the process mixes, the more rapidly it will converge to $\rho\left(\underline{m}|\underline{d}_0\right)$.

Base Representation—The representation of the state space (i.e. defining the individual states, the neighborhood structure, and the transition probabilities) is critical to the overall effectiveness of the simulation. In those cases where traversal of the state space is expensive and/or time intensive (e.g., computationally intensive forward models), the proportion of states occupying the bulk of the distribution must be kept to a minimum; this is the key tenet of the stochastic engine method. For the underground transport problem, the lithology model (generated by the TSIM code) used to generate our proposal samples is extremely efficient in this respect. We use a geostatistical model to generate the "prior" spatial distribution of physical properties (resistivity, permeability, etc.) for each iteration in the MCMC. Given that resistivity and permeability depend on lithology or facies (rock categories with distinctive characteristics), we have employed a categorical geostatistical simulation approach. The state space is defined to consist of those combinations of voxel-level lithologic labels that are consistent with our prior spatial distribution. The main advantages of this approach are: (1) data are often categorical (e.g. lithologic descriptions), (2) geologic insight on the spatial characteristics of geologic systems (e.g., facies models) can be exploited, and (3) a very large proportion of the information known about the system can be represented very compactly using only a few lithologic categories.

Resolution is critical; if the resolution is too fine or involves a high dimensional state vector, the convergence may be slowed beyond practical limits. If the resolution is too coarse, then the simulation results may prove too diffuse to serve as a basis for inference. An approach we will implement in the final year for resolving the most complex systems is to adopt a multi-resolution method. Such an approach involves examining the state space hierarchically across a variety of levels of resolution. Another method for managing state cardinality is to employ meta-states that serve as pseudo-equivalence classes, which map highly similar states into single representative entities. We are evaluating several approaches designed to define and "bin" the meta-classes that take into account the spatial characteristics of the states.

Staged MCMC—The likelihood function computations required of all proposed state transitions present another serious operational obstacle when computationally intensive forward models are involved. This follows from that fact that a likelihood $L(\underline{m})$ is essentially a probabilistic measure of the fit between the predicted data based upon model $\underline{m}$ and the corresponding observational data. Hence, nontrivial likelihood computations will require the generation of predictions via some forward model. To mitigate this problem, we implemented an innovative method that effectively reduces the number of forward model runs. It is called the "cascade" or "staging" rule and is applicable when we have observational data of differing types like lithology, ERT and flow. In these cases, the errors in prediction are often independent and hence the total likelihood expression factors into distinct terms – one for each data type. Hence, for the above example the total likelihood expression factors as follows,

$$L_{total}(\underline{m}) = L_{lith}(\underline{m}) * L_{ERT}(\underline{m}) * L_{flow}(\underline{m})$$

This probabilistic structure can be leveraged to streamline the transition process employed by our algorithm. Specifically, we have proven that performing a single Metropolis transition step (the 3 step method listed above) that uses the entire likelihood expression $L_{total}(\underline{m})$ is equivalent to performing a sequence of Metropolis steps – one for each term in the above expression. This staging form of our algorithm allows the processing order of the likelihood terms to be arranged according to increasing runtime complexity of the corresponding forward model. Hence, once a model is proposed by the prior distribution, the forward model is solved initially for the first data type alone (step 1). At this juncture, the proposed model may be rejected or accepted (steps 2 and 3). If the decision is to reject the proposal, then the forward models for the other data sets are not executed. The prior distribution simply proposes a new state and the decision process begins anew with the first data type. If the decision at this stage is to accept the proposal, the next data set is considered, its corresponding forward model is run and a decision to accept or reject is made based upon its likelihood. This continues through all of the different types of data until the proposal either is accepted at all stages or is rejected at one stage and starts over at the beginning of the sequence.

Development of the Engine Software

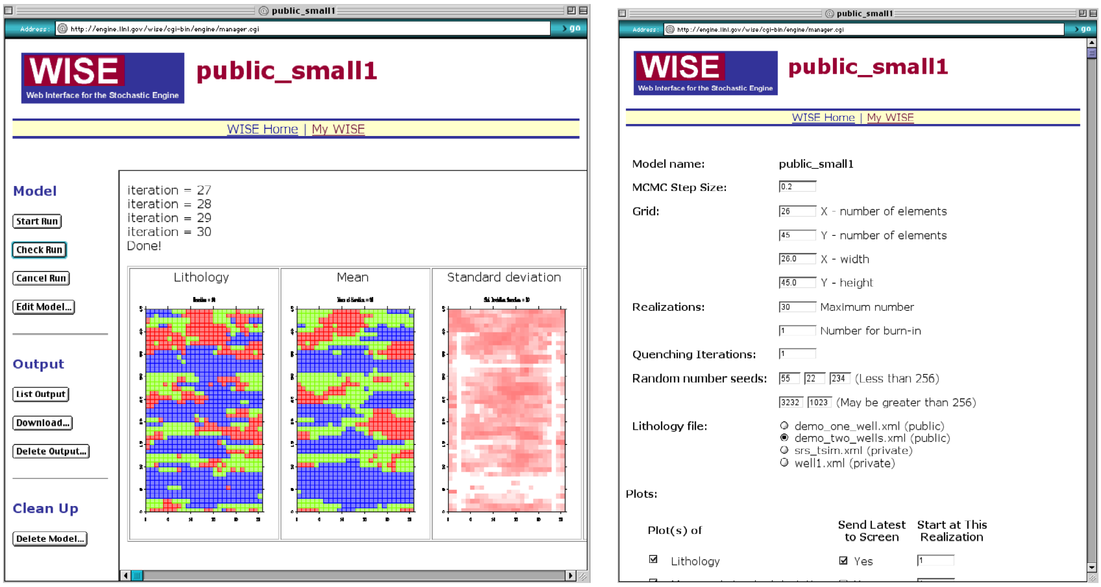
The stochastic engine system consists of six major software components: (1) the web interface, (2) the engine driver, (3) the stochastic search algorithm, (4) the data analyzer, (5) the proposal sampler suite, and (6) the forward model suite.

Stochastic Engine User Interface

The stochastic engine has, at present, two user interfaces. The first is a CLI (Command Line Interface) based on the Python language interpreter and a set of user-modifiable Python classes. The interpreter and classes combine to allow a user to specify all relevant settings. Such settings include forward models, grids, lithology data, and output files. The Python classes give users access deep into the internal workings of the stochastic engine, important for development efforts.

The second user interface is a Web-based interface (Figure 2) with the working name of WISE (Web Interface for the Stochastic Engine). WISE is a first effort at allowing off-site access to the Stochastic Engine while keeping the Stochastic Engine itself on-site. Requirements for this interface:

- It must run on a wide variety of computer platforms.
- It must shield the user as much as possible from the stochastic engine's underlying programming structure and file formats.
- It must be a GUI (Graphical User Interface), at least to the point of using forms-based input and providing access to graphical output.
- It must require the user to log in to his or her personal account.



http://engine.llnl.gov/wise/cgi-bin/engine/manager.cgi

```
iteration = 27
iteration = 28
iteration = 29
iteration = 30
Done!
```

Model name: public_small1

MCMC Step Size: 0.2

Grid: 26 X - number of elements

45 Y - number of elements

26.0 X - width

45.0 Y - height

Realizations: 30 Maximum number

1 Number for burn-in

Quenching Iterations: 1

Random number seeds: 55 22 234 (Less than 256)

3232 1023 (May be greater than 256)

Lithology file:
○ demo_one_well.xml (public)
● demo_two_wells.xml (public)
○ srs_tsim.xml (private)
○ well1.xml (private)

Plots:

Plot(s) of	Send Latest to Screen	Start at This Realization
☑ Lithology	☑ Yes	1

Figure 2. Screen images of the web-based interface. The display updates dynamically as the analysis proceeds, providing similar user feedback to the command-line interface.

The ubiquitous presence of Web browsers on Internet-connected computers made a Web-based interface a good solution.

WISE supplies each remote user with a customized interface. It allows users to define and edit models, initiate and monitor runs, and manage their own input and output files. WISE filters their access to the stochastic engine. The next stage of WISE and the I/O portions of the stochastic engine will jointly:

- Create a database to provide additional user services and improved software communications. The database may be written to or queried by WISE, the stochastic engine, and other programs such as post processors or system monitors.

- Add interfaces to additional forward models.
- Add methods to preview data files and post process output files.
- Use XML (eXtensible Markup Language) for structuring configuration files and data files, thus allowing the files to be more easily used for a variety of purposes and by a variety of programs.

Any distributed processing, where other computers on the network do all or part of the modeling work, will be managed by the stochastic engine. The user interfaces will only need to interact with the part of the stochastic engine that runs on the local machine. The database mentioned above will eliminate the need for the user interfaces to deal directly with the distributed machines.

The Engine Driver

The stochastic engine driver orchestrates the overall execution of the stochastic engine. It reads the user input, manages interprocess communication for running multiple processes on different machines, such as those in workstation clusters, and manages the postprocessing of output data, such as convergence monitoring and graphics. The following is a list of current and planned capabilities.

- Calls the appropriate stochastic search algorithm
- Drives the proposal sampler from the proposal sampler suite
- Manages the execution of the appropriate data analyzer component
- Drives forward models from the forward model suite locally or across the network (using Unix fork, telnet, socket servers, message passing)
- Runs independent chains on separate processors
- Passes input and output data between processors (using ftp, sockets, and message passing protocols)
- Performs communication with the WISE web interface
- Does scheduling of remote jobs

Note that the application of the engine to a new problem requires integrating a proposal sampler and a set of forward models into the engine. This task is made straightforward through the use of the Python programming language and object-oriented methodology (Figure 3).

We have the capability to run independent chains on different processors in local workstation clusters and the ability to manage parallel jobs spun off by forward models. These capabilities are at a prototype stage and further development will be required in order to run on a large variety of machines. Graphics can be performed on the user’s workstation, as opposed to the remote machine, to eliminate bottlenecks in data transfer. Background rendering can be done on remote machines.

In the future, management of remote jobs will be enhanced by a smart scheduler that will send jobs to a variety of non-local platforms (e.g., Teracluster, ASCII machine) depending on necessary CPU and memory resources required. The need to manage the resulting datastream will also need to be addressed. Two-way communication between the driver and the web

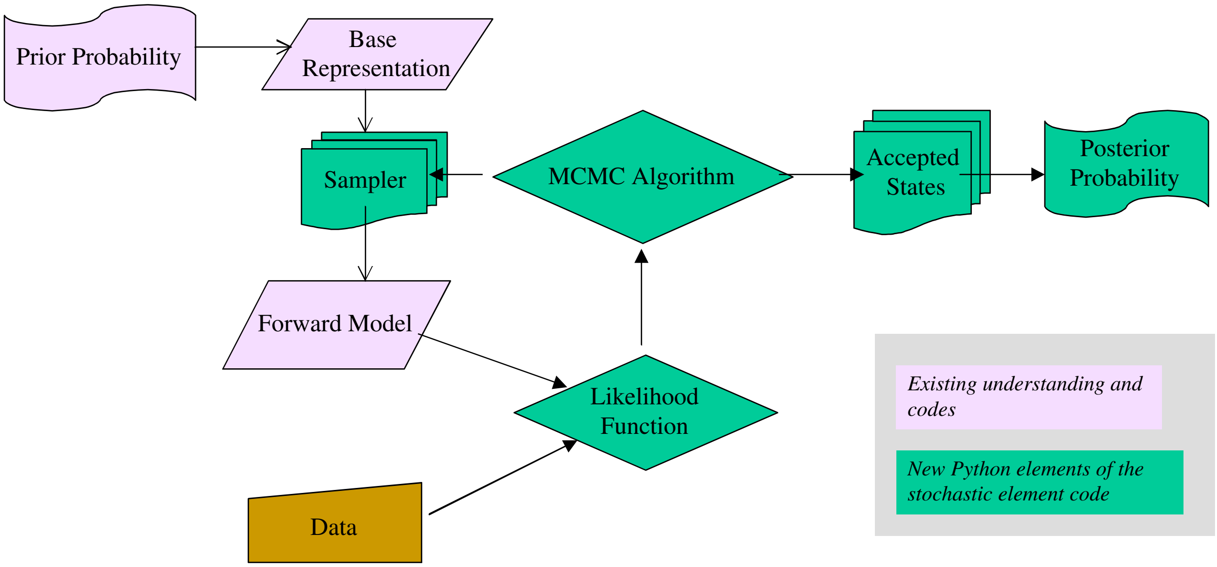


Figure 3. All stochastic engine applications share a number of software components. Existing codes are incorporated with a minimum of effort.

interface will be implemented to allow the user to interrogate the progress of the stochastic simulation.

Stochastic Search Algorithm

The stochastic search algorithm rejects or accepts realizations generated by the proposal sampler according to a probabilistic rule. The following algorithms have been implemented or are planned.

- Single-stage Metropolis and Mosegaard algorithms
- Multiple-stage Metropolis and Mosegaard algorithms
- Nitao-Hanley averaging algorithm
- Multiresolution algorithms
- Adaptive algorithms

The Nitao-Hanley averaging algorithm (Nitao and Hanley, 2001a,b) and several adaptive algorithms were developed to accelerate convergence. These algorithms as well as multiresolution algorithms will be implemented into the engine and tested.

The Data Analyzer

Real-time postprocessing of the engine output and its graphical representation is needed for monitoring the convergence of the simulation and for analyzing the statistical results. The sequence of realizations resulting from an engine simulation may also be saved for off-line postprocessing. The data analyzer:

- Performs on-line 2D and 3D graphics (using the Plotutils and VTK visualization libraries)

- Computes on-line convergence diagnostics
  - 2D convergence diagnostics for geological systems
  - 3D convergence diagnostics for geological systems
  - Convergence diagnostics for parameters characterizing chemical systems
  - Convergence diagnostics for parameters characterizing atmospheric transport
- Conducts metastate and clustering analysis

Three-dimensional graphical visualization is incorporated into the engine through an interface with the VTK visualization library. Jobs are spawned on the user's workstation to avoid graphics being a bottleneck to the progress of the simulation. Convergence diagnostics for 3D geological problems are being tested. Convergence diagnostics for new applications such as chemical and atmospheric systems will be implemented, and new visualization options will be added.

The Proposal Sampler Suite

The proposal sampler suite is comprised of a set of application-specific programs, whose purpose is to generate random realizations obeying a specified prior probability distribution. The stochastic MCMC search algorithm uses the generated realizations. The programs may already exist written in some other language such as Fortran or C++ and executed as processes spawned by the engine driver, or they can be implemented as Python modules. Current and planned proposal samplers are:

- TSIM geological lithology generator (Fortran program)
- Spectral lithology generator (Python module)
- Subsurface object location generator (Python module)
- Geochemical reaction parameter generator (Python module)
- Chem-bio process plant realization generator
- Atmospheric source generator

The Forward Model Suite

For a given application, the MCMC algorithm requires a forward model for each common set of data measurements. The purpose of a forward model is to take a realization generated by the proposal sampler and compute the prediction of the measurement. The MCMC algorithm then compares the predicted data measurement with the actual measurement in order to decide whether to accept the realization. Current and planned forward models are:

- OC4 2D Electrical Resistivity Tomography (ERT) model (Fortran program)
- Multibh 3D ERT model (Fortran program)
- NUFT well test model (C++ program)
- NUFT geochemical reactive transport model
- NUFT tracer model
- ASPEN chem-bio process plant model
- ARAC atmospheric transport models

Finer-grained parallelization of forward models may be required, depending on the size of the problem.

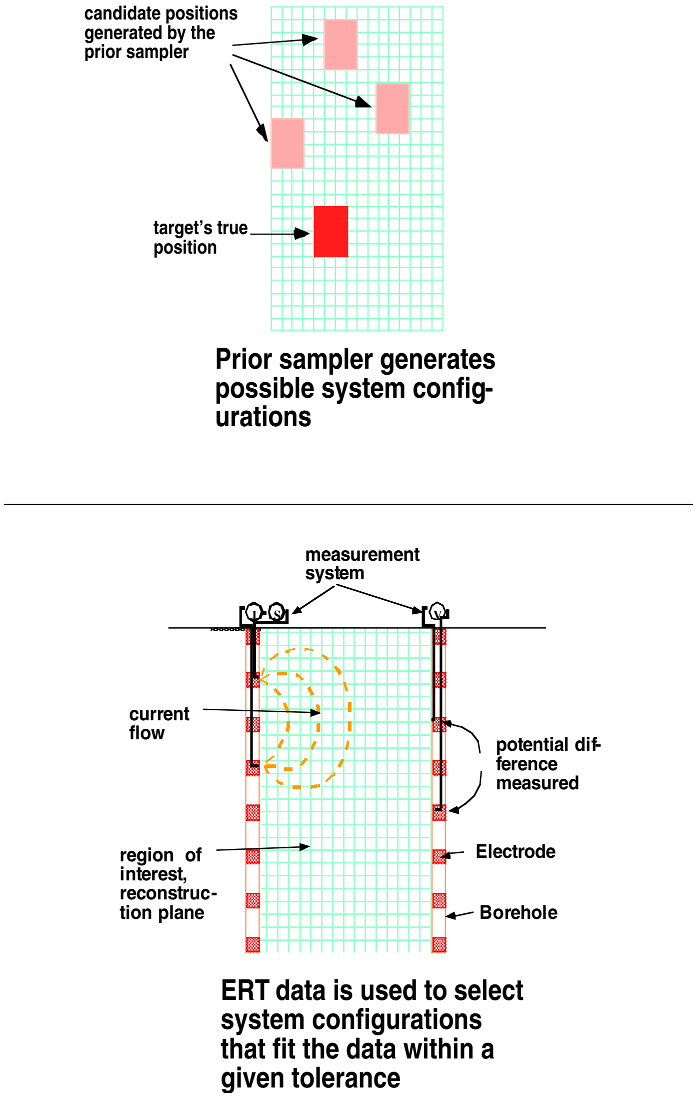
General Understanding

Example: Blob Problem

In order to have a test problem for imaging applications with a state space structure that is readily diagnosed, we developed the “Blob” problem (Figure 4). A target of fixed size and properties is assumed to exist in a homogeneous matrix. The engine’s job is to locate the target; uncertainty arises from the inherent error and non-uniqueness in the measurements. The relationship of the base representation (a series of proposed locations for the target) and the posterior distribution is straightforward for this problem; the posterior is the weighted sum of all the accepted locations of the target.

Comparison of MCMC and Deterministic Approaches

The electrical resistance tomography (ERT) inverse problem can be solved using stochastic or deterministic approaches. Both approaches have to contend with the highly non-linear relationship between the measured resistance values and the resistivity of a region. Also, both approaches face the problem of electrical equivalence (also referred to as non-uniqueness: the measured values represent equally well an infinite number of resistivity models). In general, the stochastic engine solution will require substantially longer computing times but offers the capability of jointly inverting orthogonal data sets thereby improving the fidelity of the result.


candidate positions generated by the prior sampler
target’s true position
measurement system
current flow
potential difference measured
Electrode
Borehole
region of interest, reconstruction plane

Prior sampler generates possible system configurations

ERT data is used to select system configurations that fit the data within a given tolerance

Figure 4. The blob problem assumes a fixed-shape object exists in a homogeneous matrix.

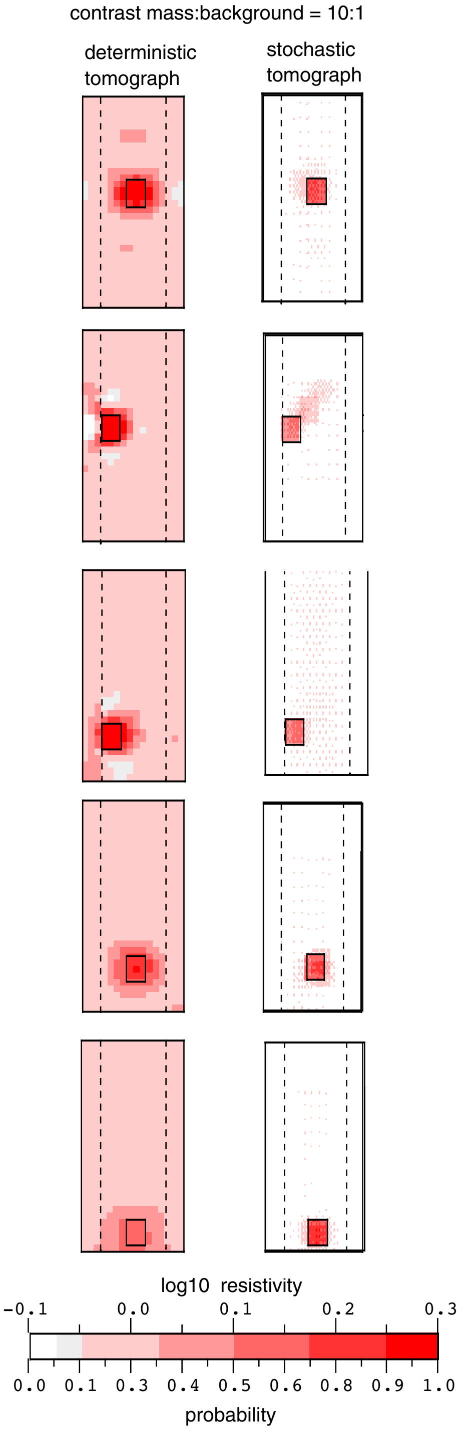


Figure 5. Comparison of deterministic and probabilistic (stochastic engine) solutions to the blob problem.

The deterministic inverse approach seeks to find a unique, robust solution to the inverse problem. Direct linear inversion is not possible because it only applies to linear problems and the ERT problem is generally underdetermined. The inverse algorithm uses a smoothness-constrained, least squares, iterative approach that searches for a single model that fits the data within a specified tolerance. Using this approach it is possible to obtain meaningful “smoothed” parameter models even when the problem is underdetermined. Also, this approach results in a unique solution that is relatively insensitive to the starting model assumed by the inversion process.

We have used the blob problem to compare the attributes of the stochastic engine and deterministic ERT approaches. The target considered consists of a mass (10 ohm-m) embedded in homogeneous background (1 ohm-m).

A comparison of the stochastic and deterministic tomographs is shown in Figure 5. The column of images on the left of the figure shows the deterministic tomographs obtained for a variety of target positions. The location of the target is shown by the black rectangle and the dashed vertical lines indicate the locations of the electrode arrays. The column of images on the right show the corresponding stochastic tomographs; these show the posterior probability distribution produced by the stochastic engine.

A comparison of the images in Figure 5 suggests that both approaches do a reasonable job of mapping the gross target location. The deterministic tomographs show a map of the electrical resistivity of the system; the highest resistivity values indicate the target location. The stochastic results show the probability that the target is present at a given location; the highest probabilities indicate the most likely target locations.

The most striking difference between the two approaches may be that the shapes and sizes of the "truth" model are substantially more distorted in the deterministic tomographs. The distortion is directly attributable to two factors:

(1) The smoothness constraint employed by the deterministic approach "smears" the resistivity values between adjacent elements thereby producing models that have reduced contrast and exaggerated extent.

(2) The size of the solution space is much smaller for the stochastic engine approach due to the constraints on the stochastic search. The base sampler used by the stochastic engine only considers models where the target location is variable but the size and contrast of the target remain fixed. In other words, the base sampler takes advantage of prior knowledge regarding the target's size and shape. The MCMC inversion has fewer degrees of freedom, and this results in better target resolution. Alternatively, the deterministic inversions have to discriminate between many more models because target size and contrast are unconstrained; there is no way to incorporate prior knowledge.

Because of the spatial nature of the blob problem state space, we can map the shape of the likelihood surface, as in Figure 1. Transects of the likelihood surface are shown in Figure 6, for a wider version of the blob problem. For the line through the center of the box (blue diamonds), the surface is very steep near the actual location, showing that there is very little uncertainty associated with finding the blob in this problem. If there were more than one blob in this example, the stochastic engine result would be far superior to the deterministic inversion.

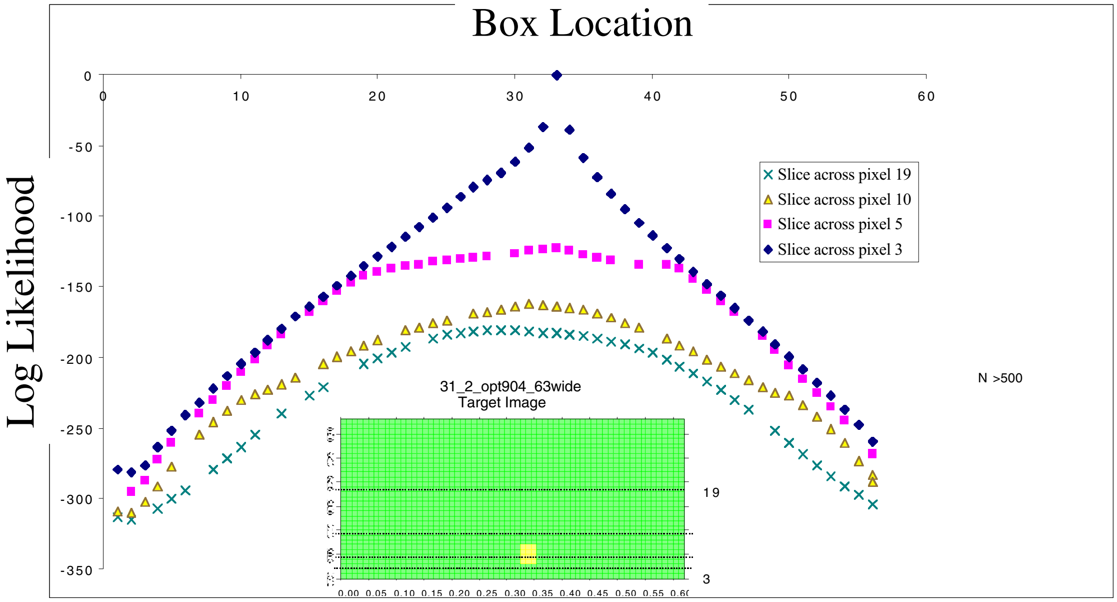


Figure 6. Likelihood values for the blob problem can be easily plotted as a function of blob location. The Metropolis/Mosegaard search algorithm rapidly finds a singular maximum of this sort: the interesting question is to accurately include error to establish a confidence interval.

Example: Savannah River A/M Area Imaging

The SRS Model and Base Representation

The Savannah River Site (SRS) is situated upon fluvio-deltaic (river and delta) deposits of several hundred meters thickness. The hydrostratigraphy consists of a stacked sequence of aquifers and aquitards related to ancient depositional environments. The aquifers predominantly consist of sandy fluvial deposits, whereas the aquitards predominantly consist of clayey overbank (flood) deposits. However, both the aquifers and aquitards consist of a complex three-dimensional architecture of materials ranging from clay to gravel.

In our initial applications of the stochastic engine to SRS, we focused on the shallow subsurface of a small subregion called the A-14 outfall, where a waste stream was historically discharged into the unsaturated zone of a shallow aquifer. In this setting, contaminant fate is dictated by the complex spatial distribution of the small proportion of clayey zones. We obtained lithologic and geophysical logs, cone penetrometer data, and geologic cross-sections from nearby boreholes. We compiled lithologic data from eight nearby boreholes, and also completed a preliminary geostatistical analysis of the site, which was used to initiate our prior lithology model for the stochastic engine.

The stochastic simulation code "TSIM" is used to generate base representations of lithology (the prior). TSIM (Carle, 1996; Carle et al., 1998) is the only geostatistical simulation code that accurately honors the spatial variability model for multiple lithology problems. Figure 7 shows three "realizations" generated by TSIM for the Savannah River Site application. The four colors represent different lithological categories: gravel (red), sand (yellow), clayey sand (green), and clay (blue). Each realization exhibits a similar pattern of spatial variability that is consistent with borehole data and geologic descriptions of the site. TSIM honors "hard" data, such as lithologic data at boreholes. In each realization, hard data are honored for the borehole on the right side (indicated by the black line).

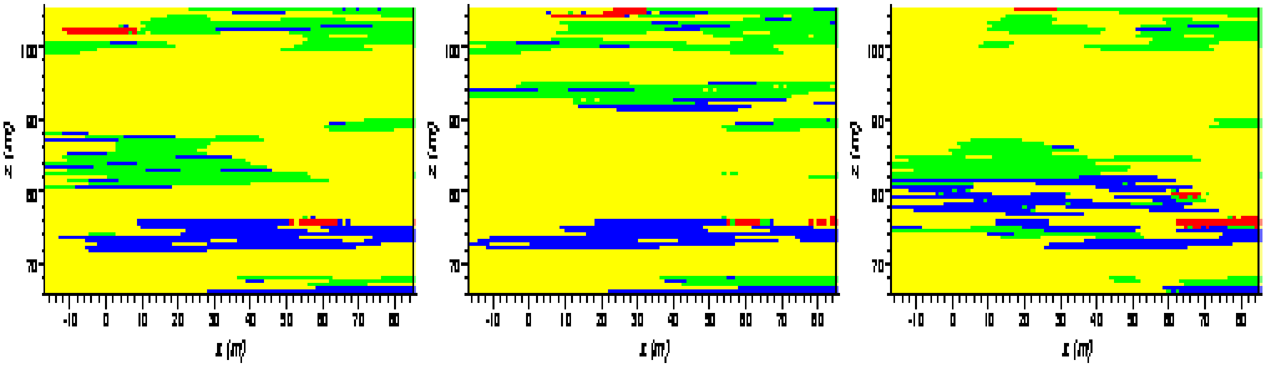


Figure 7. Three realizations of the Savannah River Site generated by TSIM. Borehole data are honored on the right side.

Several important improvements have been made to TSIM for this initiative. Run time for TSIM has been decreased by a factor of 10+. Four new capabilities have been added to TSIM:

- Realizations can be generated that are similar to previous realization, as required by the MCMC algorithm.
- Using precalculated cokriging weights enhanced the efficiency of the program.
- Soft data, such as electrical resistivity logs, cone penetrometer data, or other forms of indirect data, can be used to condition the realizations.
- Prior knowledge of "nonstationarity" of lithology proportions, e.g. information indicating that a certain lithology is more likely to occur in a certain area, can be considered.

The ability to regulate similarity of realizations provides the "stepsize" control crucial to implementation of the MCMC algorithm. The ability to use soft data opens many opportunities for further conditioning of the realizations, considering that most geologic data are uncertain. The ability to incorporate nonstationarity enables TSIM to integrate geophysical imaging and geologic interpretations of localized variations in stratigraphy as prior information. Moreover, the nonstationarity capability enables the use of previous stochastic engine runs as prior information encapsulated by probability maps.

TSIM generates stochastic simulations by a two-step process using the algorithms of "sequential indicator simulation" (Deutsch and Journel, 1992) and "simulated quenching" (Carle, 1997). For both steps, the model of spatial variability is a three-dimensional spatial Markov chain (Carle and Fogg, 1997). In the sequential indicator simulation step, the algorithm visits each cell of the realization along a random path and uses cokriging, a form of linear regression, to estimate the probability that a particular lithology occurs at that cell. The factor of 10+ speed-ups has been achieved by recognizing that solution weights to the cokriging equations are identical for the same random path. By storing the cokriging solution weights from the first realization and maintaining the same random path in subsequent realizations, probability estimates for subsequent realizations involve only computation of weighted sums. Previously, the sequential indicator simulation step required the bulk of the run time; now it is faster than the quenching step.

The sequential indicator simulation step provides the "initial configuration" for the simulated quenching step. Simulated quenching is the "zero-temperature" case of simulated annealing, where perturbations that reduce the objective function are accepted with a probability of 1.0 (versus < 1.0 for simulated annealing). The simulated quenching step is implemented by visiting cells along a random path and testing whether a change in lithology will reduce the objective function, which measures the difference between the spatial variability of the realization and the model.

The new capabilities listed above are implemented by modifying the cokriging linear systems of equations, the cokriging estimate, and the frequency of visitation of cells for the simulated quenching step. Figure 8 shows an example using soft data, which are located along the dashed lines on each realization. The soft data tighten the probabilities that certain lithologies occur at the soft data locations.

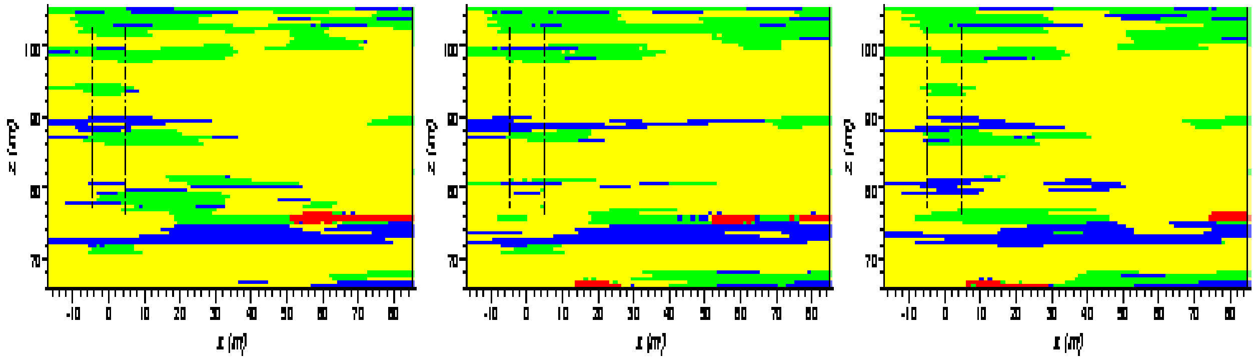


Figure 8. Three realizations conditioned by both hard (solid line, at right) and soft (dashed lines, left and middle) data.

Figure 9 shows an example using a stepsize of 0.5 to create a series of realizations where the new realization is similar to the previous realization. The similarity is controlled by the stepsize, where 0.0 exactly reproduces the previous realization, and 1.0 has no effect.

Figure 10 shows an example where nonstationarity of lithology proportions is considered. The top half of the realizations assume proportions of gravel = 0.011, sand = 0.930, clayey sand = 0.056, and clay = 0.003; the bottom half assumes proportions of gravel = 0.000, sand = 0.541, clayey sand = 0.256, and clay = 0.203.

The new capabilities added to TSIM have been essential to application of the stochastic engine to “real world” subsurface problems. TSIM is now vastly superior to any geostatistical lithology generator available today.

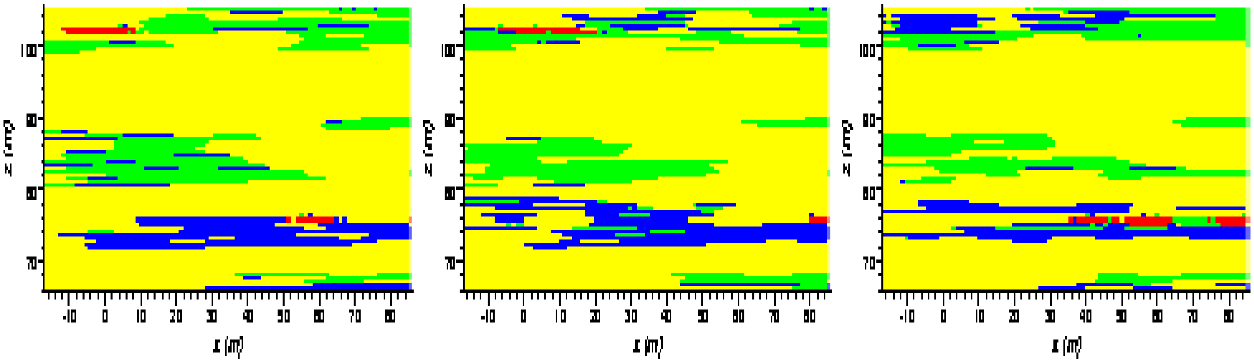


Figure 9. A series of three realizations where stepsize controls similarity of one realization to the next.

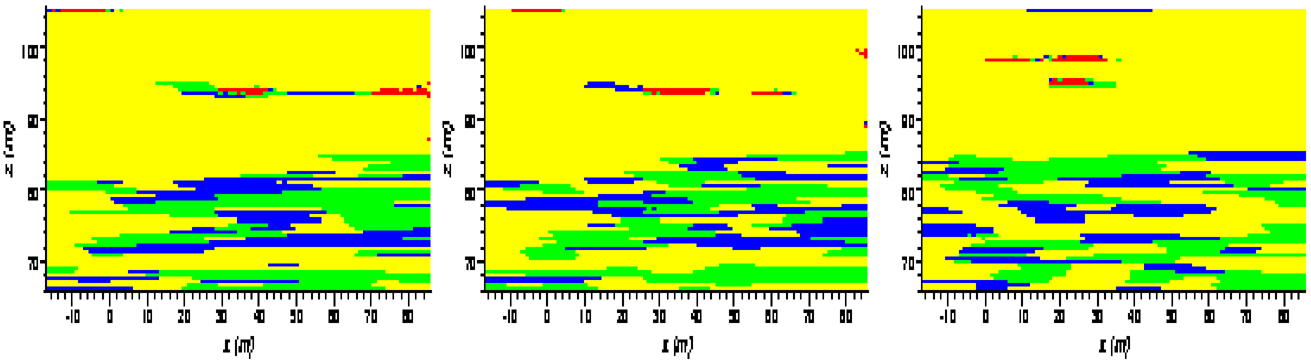
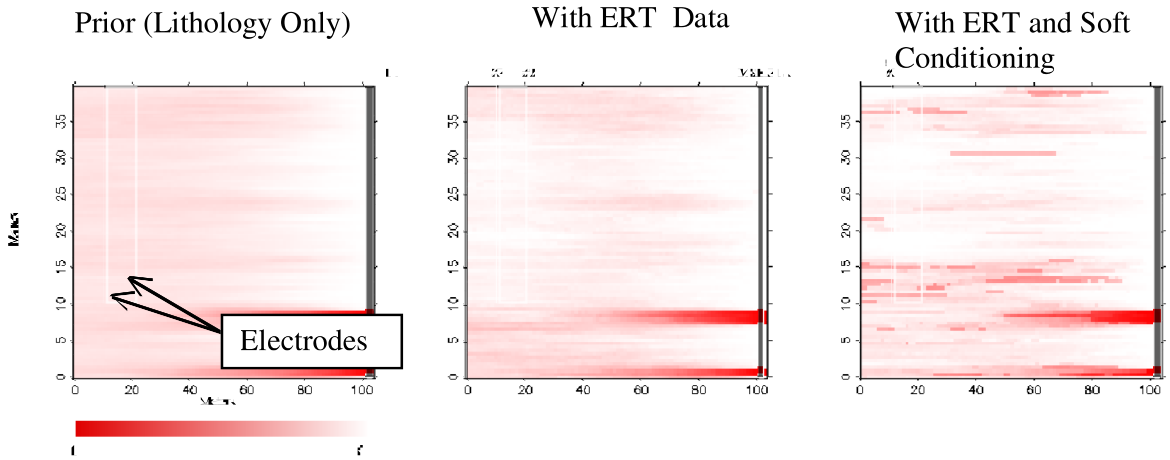


Figure 10. Example of nonstationarity, where different lithology proportions are given for the upper and lower halves of the realizations.

ERT and Soft Conditioning from Cone Penetrometer Data

We have analyzed 2D ERT collected at the Savannah River Site A/M outfall area as part of an ongoing DOE environmental characterization project. The purpose of this work is to demonstrate the performance of the stochastic engine using real data from a site where improved data interpretation methods can make a difference in remediation schedule and cost.

Results: Pixel-Wise Probabilities. Figure 11 shows stochastic tomographs generated by the stochastic engine. When working in lithology space, one way to visualize the summation of the accepted states is in terms of the pixel-wise probability of each lithology type. The left image of Figure 11 shows the posterior probabilities for the clay category (on a pixel by pixel basis) produced by the lithology generator. The vertical white lines show the electrode array locations. The lithologies on the right side of each image are "hard-conditioned" by lithology information observed along a well (well location indicated by a dark, wide vertical line). This means that wherever clay was observed along the well, the probability for clay at that location is 1.0.


Prior (Lithology Only)

With ERT Data

With ERT and Soft Conditioning

Electrodes

Figure 11. Probability of clay at any pixel in the SRS A/M basin. (Left) The prior distribution (before stochastic engine analysis). (Middle) With ERT data included in the in analysis, layering can be discerned near the electrodes. (Right) Using additional data (electric logs) from the electrode installation, clear layering is apparent.

The left image in Figure 11 honors the global lithologic tendencies at SRS but does not use any data collected locally near the electrode array location. This is the “prior” for this analysis; the information conveyed by our lithologic knowledge alone. As a result, the tomographs are “flat” near the electrode arrays, i.e., they show no evidence of distinct layering. This means that, far away from the well on the right where we absolutely know the lithology, there is a very nearly equal probability for a given layer to be located anywhere within the image. The information provided to the MCMC algorithm is insufficient to position the layers in space. The middle image shows the posterior probability calculated when the lithology sampler and the ERT data are used together. This image honors the global lithologic tendencies as well as the ERT survey data. The posterior distribution now shows evidence of layering, although not very clearly. Interestingly, the ERT data indicates that there is less clay in the vicinity of the electrodes than is generally present in this area.

The influence of soft conditioning and ERT data can be seen in the rightmost image of Figure 11 The right image shows the posterior probability calculated when the lithology generator, the ERT data and the electrical well logs are used together. In this case, the electrical logs are used to “soft-condition” the realizations (i.e., introduce a bias) along the electrode array locations. The soft-conditioning forces the lithology generator to honor global lithology data and local lithology inferences based on the electrical logs.

In order to facilitate the incorporation of soft data, we created a Bayesian predictive time series statistical tool. This tool can be used to incorporate other data types in future problems. To implement the soft conditioning methodology, lithologic probability distributions were estimated at each pixel in the given resistivity well. This was accomplished by first building a collection of Bayesian time series models which incorporate measured resistivity data (from electric borehole logs) and general lithologic/resistivity correlative properties to generate class conditional predictive distributions of resistivity. Then, for each pixel adjacent to the well, the likelihood of its measured resistivity, conditional on each possible lithologic class, was computed using the assembled time series models. These likelihoods were combined with the prior lithologic distribution using Bayes’ Theorem to produce a posterior distribution of lithology at each pixel. These posterior distributions constitute an updated probabilistic representation of our current state of knowledge concerning the lithologic identity of each well pixel.

The rightmost image of Figure 11 image shows evidence of distinct clay layers. Note that the probabilities have increased at distinct locations thereby pointing to likely presence of clay layers. This type of information is crucial to understand the transport of subsurface contaminants at SRS because the clay layers control transport behavior. The results in Figure 11 suggest that the stochastic engine can be used successfully to image spatially complicated targets such as heterogeneous layers in geologic environments.

The probability tomographs are the key product of the stochastic engine. They distill information from various orthogonal data sets and from thousands of forward calculations thereby providing an unprecedented wealth of data about the natural system. The probability tomographs are particularly useful for decision-making because, in addition to spatial data, they indicate probability distributions. As result, the tomographs can be used to infer multiple geologic configurations with similar posterior probabilities. This is particularly useful in earth science problems because most geologic data are uncertain.

Adding Information: Using the Result to Start the Next Analysis. The posterior probability plots shown above record the pixel-wise probability of occurrence of a particular lithology. This is useful for visualization of the results, but it can be used in a much more powerful fashion. By using the new capability of soft-conditioning the TSIM base representation, we can constrain a future analysis—the posterior distribution becomes the prior for the next application. TSIM provides the spatial information to decode the pixel-wise probabilities into lithologic realizations with complex spatial relationships. For instance, another well could be added at the A/M area and a new analysis conducted, much more efficiently. The new state space is much smaller, as it only contains configurations that are consistent with the previous analysis. For extremely complex analyses using supercomputer resources, this will provide a dramatic increase in efficiency. Complex runs are very effectively stored in this manner. In addition to the great effectiveness of underground imaging using the engine, we believe this will significantly simplify the task of tracking steam injection.

This posterior-to-prior capability will greatly enhance risk analysis as well. A variety of calculations that are completely unrelated to the engine can be run on the assemblage of new realizations derived from the posterior distribution. For instance, the probability of contaminant transport over many years through a site can be calculated, and then a parallel calculation made assuming a palliative remedy has been applied. We believe this is the most powerful aspect of the engine: taking the engine analysis and turning it into a knowledge base for the future, to be incremented as needed.

Application: SRS Solvent Tanks Imaging

We have also re-analyzed data collected originally as part of a steam remediation at the Solvent Tanks site at SRS. This site was near the A/M site, and we were able to use the same lithologic model. The purpose of this work was to demonstrate the performance of the stochastic engine using real data from a site where improved data interpretation methods can make a difference in remediation schedule and cost; in this case there is a direct comparison available to the traditional analysis methods that were actually used at this site. The data were provided by Integrated Water Resources (Santa Barbara, CA), the contractor responsible for the design, construction and operation of the remediation system. A plan view of the site is shown in Figure 12. The electrode arrays used to collect the ERT data are located in boreholes TM6 and TM8. The electrode array locations define a region of interest 36 m wide and 51 m deep. This ERT plane was chosen because it is in close proximity to borehole MRS 12 where samples identifying subsurface lithologies were collected.

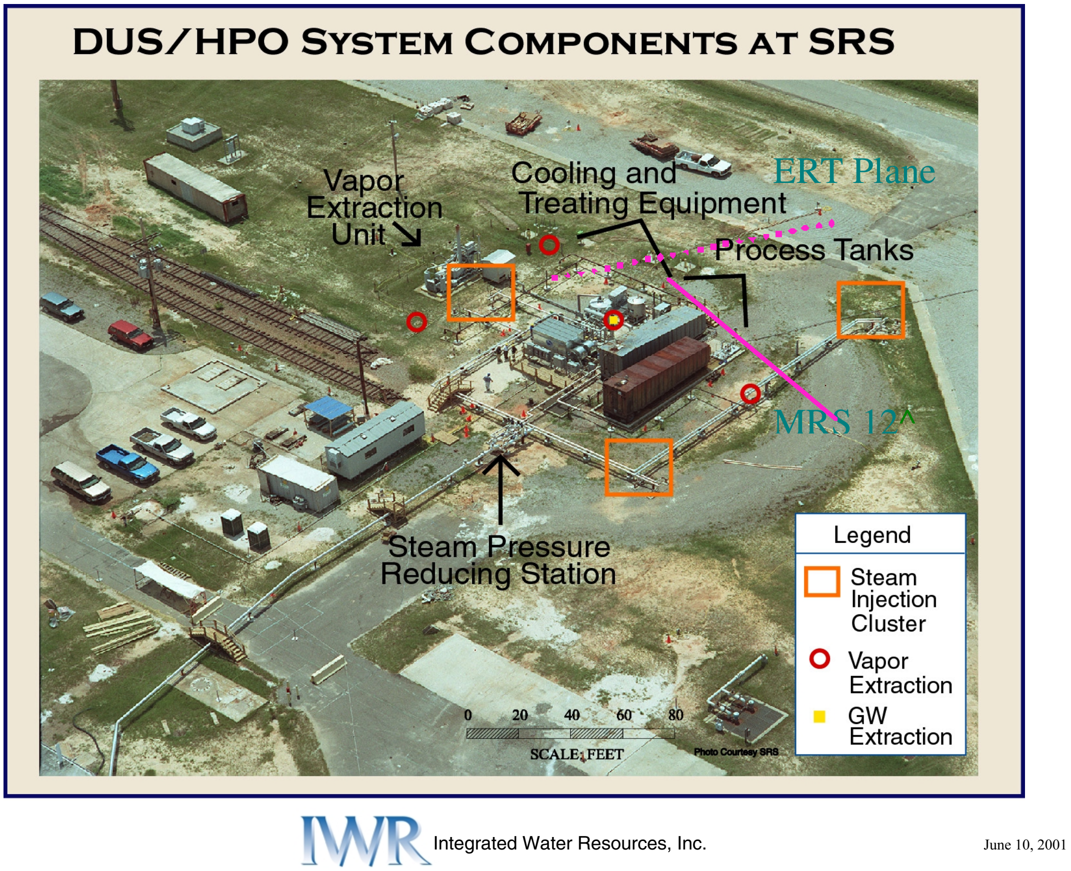


Figure 12. Plan view of the solvent tank storage area, Savannah River site. Electrode arrays located in boreholes TM6 and TM8 produced ERT data that has been processed by the stochastic engine, shown in Figure 13.

Figure 13 shows the stochastic tomographs calculated using the various data sets available. Posterior probabilities were calculated for three soil categories: sand, clayey sand, and clay. The choice and number of categories were based on analysis of lithology data collected elsewhere onsite. These lithology data served to establish global tendencies for the layering at SRS, e.g., relative proportions of the various categories, mean length and thickness for layers, and juxtapositional tendencies between layers.

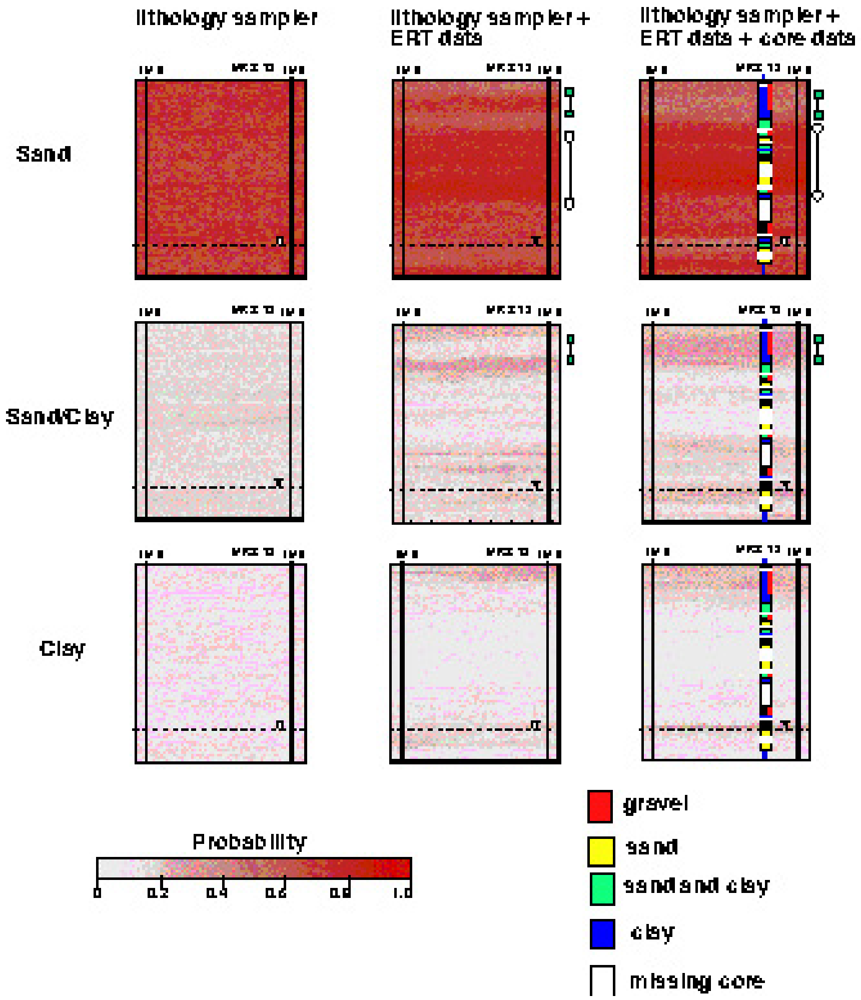


Figure 13. Stochastic tomographs from the solvent tanks site, compared to lithology determined in the MRS 12 drillhole.

The left column of images in Figure 13 shows the posterior probability calculated when only the lithology sampler is used. The vertical black lines show the electrode array locations. The horizontal dashed line indicates the water table depth. These images honor the global lithologic tendencies at SRS but do not use any data collected locally near the region of interest. As a result, the tomographs are “flat,” i.e., they show no evidence of distinct layering. The probabilities for the categories are primarily dependent on their relative volumetric proportions. For example, sand shows the highest probability because it is known to occupy the largest volume under the SRS site.

The middle column of images shows the posterior probability calculated when the lithology sampler and the ERT data are used together. These images honor the global lithologic tendencies as well as the ERT survey data. The stochastic engine now has enough information to position the layers, indicated by the increased the posterior probabilities at some locations and decreased probabilities at others. For example, the image corresponding to the sand/clay category shows clear evidence of increased probability (relative to the "base sampler only" images) near the top 25% of the image.

The right column of images in Figure 13 shows the posterior probability calculated when the lithology sampler, the ERT data and the core log data from borehole MRS 12 are used together. In this case, the local lithology data is used to "soft-condition" (i.e., introduce a bias) the realizations produced by the lithology sampler such that both local and global lithology data are honored. The lithologic categories mapped along MRS 12 are superimposed on the images. These probability tomographs honor global lithologic tendencies, local lithology information (along the axis of MRS-12), and the ERT survey data.

The results in Figure 13 demonstrate the capacity of the stochastic engine to work with real field problems that include measurement and numerical modeling errors. The results also demonstrate the capacity of the stochastic engine to combine orthogonal data sets and produce probability tomographs that honor all available data. When the orthogonal data sets are in agreement, the posterior probabilities increase as new data is added. When there are conflicts between data sets, the posterior probabilities decrease as new data sets are added.

Figure 14 compares the SRS-Solvent Tank Area tomographs obtained with the stochastic engine and deterministic approaches. The deterministic tomograph is shown on the right hand side of the figure using a gray scale color bar; the stochastic tomographs are displayed using a red scale color bar. Evidence of layering can be seen in both types of tomographs, but the layering in the stochastic engine (red scale) tomographs is much more realistic because it makes use of available information about layering patterns at SRS. The deterministic tomograph searches for smooth models that are not quite as realistic as the models proposed by the stochastic engine prior.

This comparison suggests that the stochastic engine offers some significant benefits over the deterministic approach. This is an expected result because the stochastic engine makes use of two data sets (electrical and lithology measurements) whereas the deterministic approach only uses electrical data. The stochastic engine results in more accurate images because the search of the solution space is more tightly constrained and no smoothing constraint is required. The constraint for the stochastic engine is based on plausible lithologic architectures that have realistic features identified using other site data. The stochastic engine also computes a range of solutions and their probability distribution. Consequently, alternative models can be considered, information gaps identified, and the value of collecting additional information quantified. The primary drawback of the stochastic engine is that it is computationally intensive, requiring approximately 100 times more forward calculations. The stochastic engine approach also requires some prior knowledge about the system, such as the lithologic character. However, since we usually have significant information of this kind (if from nothing else, emplacing the electrodes within boreholes generates lithologic information), this is not a disadvantage.

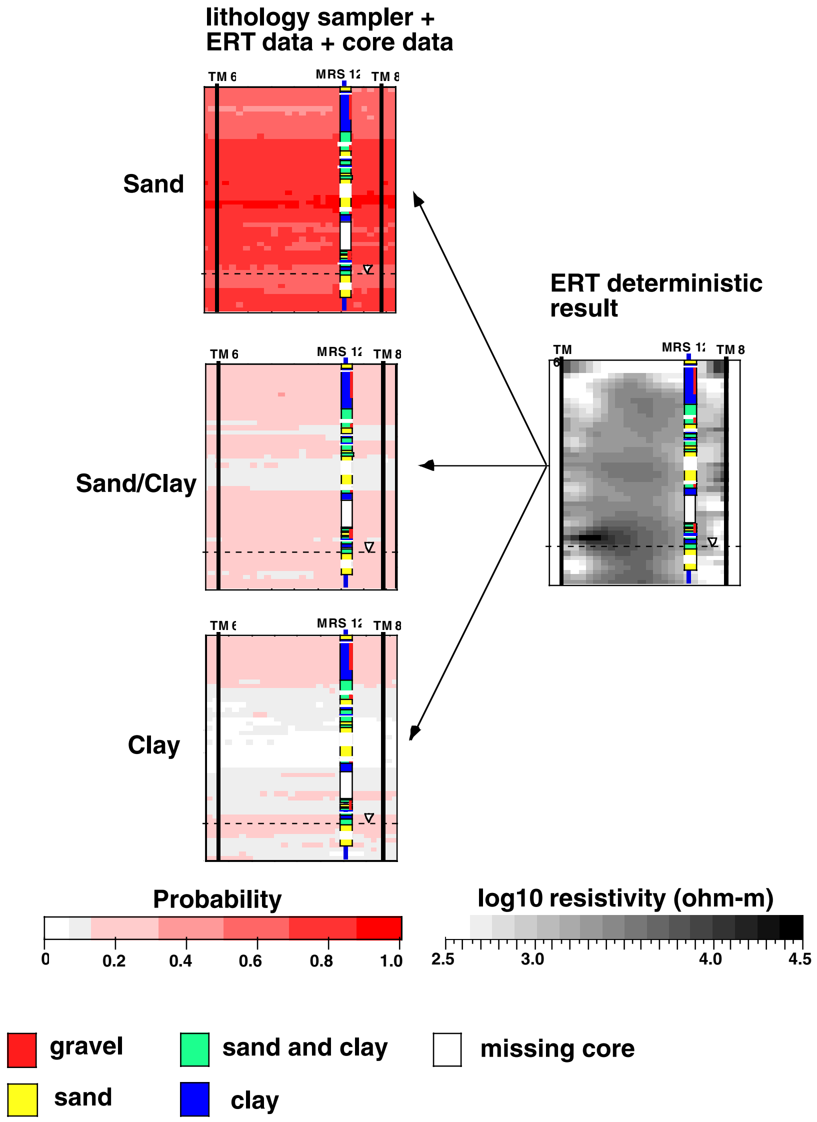


Figure 14. Comparison of stochastic and deterministic tomographs from the solvent tank storage area, Savannah River Site. The stochastic results are shown using a red color scale and the deterministic results use a gray scale.

Other Representation Spaces: Parametric Problems

<u>Application: Process-Centered Identification of WMD Activity</u>

Converting huge amounts of data into useable information is a critical national security need, and it is the primary output of the stochastic engine methodology. A natural application is identifying WMD facilities through a combination of observed facility information and atmospheric sampling information. This application would permit use of multiple kinds of data, including expert opinion, and will produce answers with quantitative confidence limits. Optimum resampling methods and locations can be identified from our analysis, and when conducted, those analyses can be combined with the original information to produce updated confidence in the products and capacity of the facility. A national security application that has been selected for study is the problem of identifying what, if any, chemical warfare agents are being produced at some location. In addition to knowing the types and quantities of agents being produced, interdiction planning is facilitated by knowledge of the particular production pathways by which the agents are manufactured. Since the answers to these questions are not usually directly obtainable, they are typically estimated from the analysis of circumstantial and indirect evidence. In this application, definitive answers are rare and reducing the range of alternative answers is an acceptable goal of analysis.

Problem. A synthetic example is being developed and is depicted in Figure 15. This (hypothetical) facility is known to produce thiodiglycol (TDG), a common dual-use chemical. It is suspected of also producing sulfur mustard, since sulfur mustard can be manufactured from TDG by adding one of several common chlorinating agents. Although the fact that TDG is produced here is public, the specific pathway is unknown. Conclusions regarding production at the site must be deduced from observations of external activity and clandestinely obtained air samples.

Base Representation. Given the scarcity of evidence that can be brought to bear on problems of this nature, it is critical that there be some objective structure against which to evaluate each item of data. For problems like the one depicted in Figure 15, the chemical process model provides that structure. This model, often referred to as a *flowsheet*, identifies the steps by which starter chemicals are transformed into products and by-products. It specifies not only the reactions involved but also post-reaction processing and waste-stream handling. Once all the steps are specified, the flowsheet can be submitted to a chemical process simulator such as ASPEN which calculates expected output quantities of all products and by-products, together with additional data such as reaction temperatures and pressures and the composition of controlled emissions. The flowsheet represents the conceptual model of the facility's activity and may or may not include a spatial representation of the components.

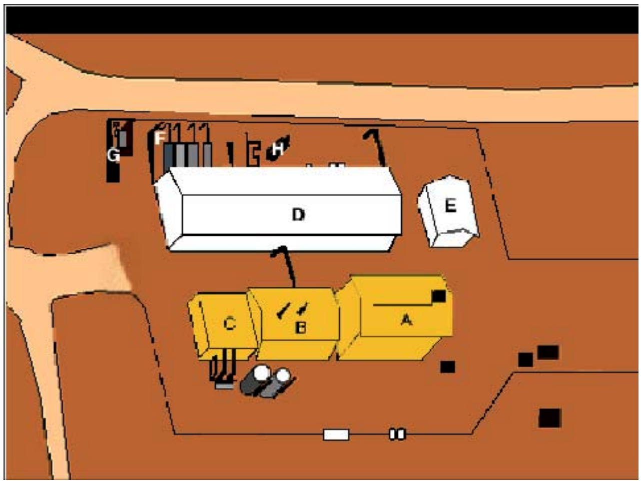


Figure 15. A hypothetical industrial site. Buildings A–C contain the main processing facilities; D and E are for storage and support.

The elements of the base representation which are manipulated for the purpose of model inversion are portrayed in Figure 16. Generating a single configuration of the flowsheet is a two-step process. First, as is shown in the left-hand portion of the figure, the product and its formulation must be determined. The TDG network shown in the figure contains two alternative formulations, path A and path B. It is not only necessary to select a path but also to determine whether the precursor chemicals are being manufactured on-site or being imported from elsewhere. The red shaded area in the figure indicates a selection of TDG manufacture via path B and that the precursors ethylene oxide (EO) and hydrogen sulfide (H_2S) will be considered the starter chemicals.

The right-hand side of the figure gives a highly simplified illustration of how the production/formulation selection is translated into a flowsheet. The starter chemicals are shown entering a reaction step from which TDG is directly obtained and after which excess H_2S is sent to a scrubber. In the scrubber, H_2S combines with caustic soda (NaOH) to produce caustic sodium sulfide (Na_2S), which is shipped to another facility for sulfur recovery. At this point, gases are also generated and vented.

Examples of manipulated flowsheet elements include continuous variables such as the available quantities of the starter chemicals and nominal variables such as the type of vessel chosen for the reactor step. Equipment alternatives have a significant impact on subsequent calculations because they affect, among other things, capacities and efficiency factors. Examples of the types of quantities calculated by ASPEN are shown in green and include both the expected quantities of TDG and Na_2S and, often more importantly, the composition of vented gases. While ASPEN also calculates pressure and temperature information, these phenomena are too diffuse to be measurable with any useful precision from outside the facility.

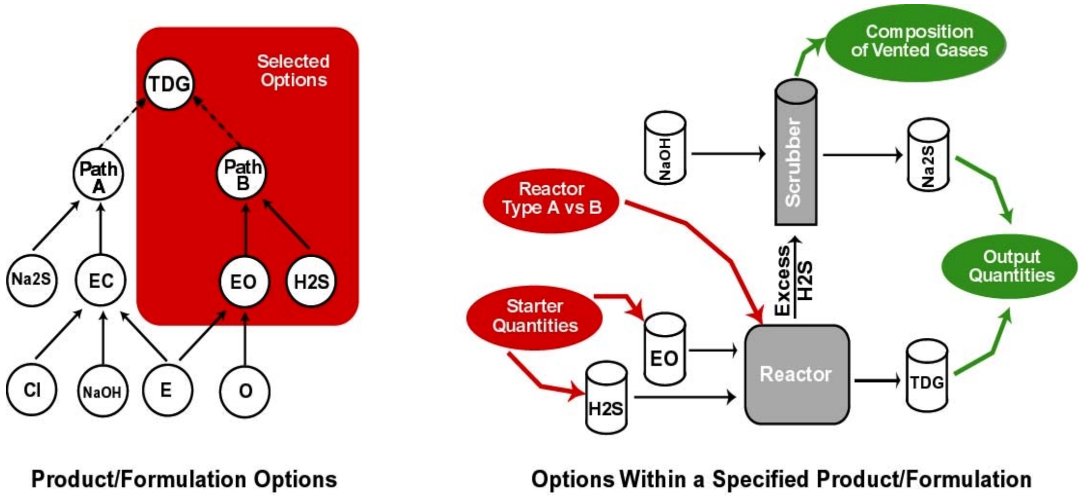


Figure 16. Base representation of the process-centered identification problem. Manipulated parameters are shown in red; outputs to be predicted and observed are shown in green. To save space, the sulfur mustard aspects of the problem are not shown.

MCMC Inversion. Figure 17 attempts to place the process-centered identification problem within the MCMC context. The *prior* and *posterior* distributions of process models are visualized as collections of flowsheets (each having a non-trivial probability of occurrence) rather than as a landscape of probabilities. The goal of the MCMC process is to winnow the large collection in the prior distribution down to a smaller set. Prior probabilities are derived mainly from common industrial practices.

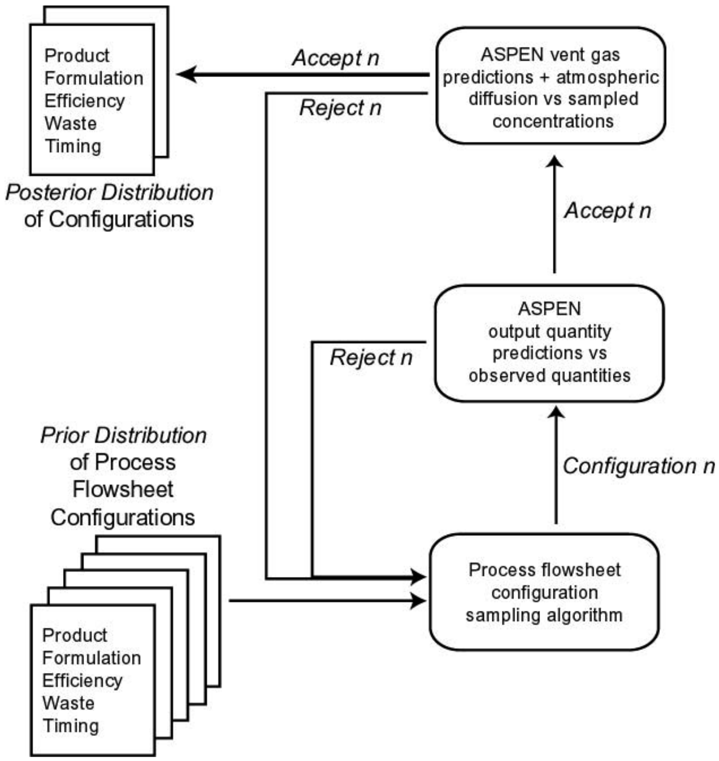

Product
Formulation
Efficiency
Waste
Timing
Posterior Distribution of Configurations
Accept n
Reject n
ASPEN vent gas predictions + atmospheric diffusion vs sampled concentrations
Accept n
ASPEN output quantity predictions vs observed quantities
Reject n
Configuration n
Prior Distribution of Process Flowsheet Configurations
Process flowsheet configuration sampling algorithm
Product
Formulation
Efficiency
Waste
Timing

Figure 17. Process-centered identification problem in the MCMC context.

The figure also shows how the ASPEN process simulator generates predictions that are compared with different data types to create a multi-stage inversion for the synthetic example. In the first stage, ASPEN calculations of product and by-product quantities are compared with "direct observations" of barrels on the loading docks. In the second stage, ASPEN calculations of the relative concentrations of vented gases at the source (i.e., the stacks shown on Building B) are combined with a simple atmospheric dispersion model to predict concentrations at various points in the vicinity. These calculations are then compared with air samples to form the second inversion stage. Although essentially the same process model is used in both stages, the independence assumption is not violated because the data sources are independent of each other.

Application: Refining a Parametric Base Representation—Reactive Transport Modeling of Plug-Flow Reactor Experiments

Reactive transport modeling is a unique methodology for numerically simulating coupled thermal, hydrological, geochemical, and geomechanical processes in the subsurface. It provides an invaluable predictive tool for both forecasting and optimizing engineered perturbations to many geologic environments of strategic and economic significance (Johnson et al., 1999). Important LLNL applications include nuclear waste disposal and geological sequestration of carbon dioxide (Johnson et al., 2001). However, in its current state of maturity, the power of this approach—explicit coupling of many complex geological processes—carries with it a daunting challenge: quantitative assessment of an integrated assemblage of model and parametric uncertainties that are, in many cases, substantial. The stochastic engine provides an innovative means of integrating and quantifying these uncertainties.

This application is fundamentally distinct from the SRS/ERT problem. Here, the base representation is defined by parametric distributions for minerals within a single lithology, as opposed to spatial distributions of lithologies having fixed parametric descriptions. Thus, the reactive transport and SRS/ERT problems define "end-member" parametric and spatial investigations.

The plug-flow reactor (PFR, Figure 18) provides a tightly constrained laboratory physical model of one-dimensional reactive flow (Johnson et al., 1998). In particular, the temperature, pressure, and flow rate are kept virtually fixed throughout the plug, the composition of infiltrating fluid and crushed rock is pre-determined, and the porosity and physical properties of-individual rock grains are measured to within a close tolerance. The time history of the effluent concentrations at the outlet side of the plug is then measured, and a post-mortem mineralogical analysis of the plug is performed.

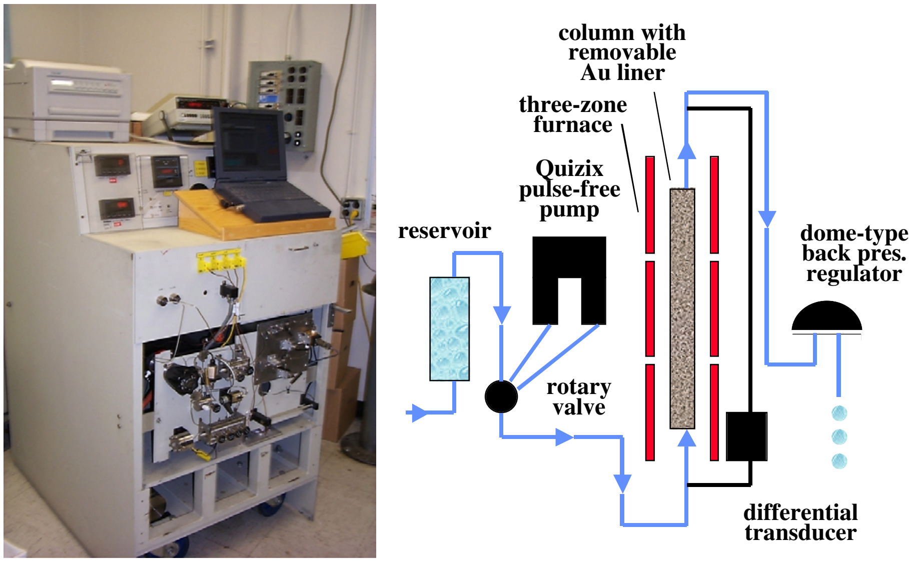


Figure 18. PFR experiments provide a well-constrained physical model of 1D reactive transport.

The essential features of observed steady-state effluent concentrations and dissolution/precipitation features can be successfully modeled using a reactive transport simulator, the NUFT code (Nitao, 1998a,b). Whether employing an iterative deterministic approach (Johnson et al., 1998) or the stochastic engine the fundamental match to be obtained is that between effluent concentrations and mineral dissolution and precipitation rates.

The tuff dissolution experiment in Figure 19 is characterized by well-constrained thermodynamic and physical parameters for primary and secondary minerals (Johnson et al., 1992), and very large uncertainties in the kinetic parameters for secondary minerals during their nucleation and early growth. In our previous deterministic modeling investigation, we adopted fixed values for the relatively certain kinetic parameters of primary minerals, then iteratively varied the more uncertain kinetic data for potential secondary precipitates until predicted and observed effluent concentrations were brought into close agreement (Johnson et al., 1998). The stochastic engine offers a more systematic and quantitative approach.

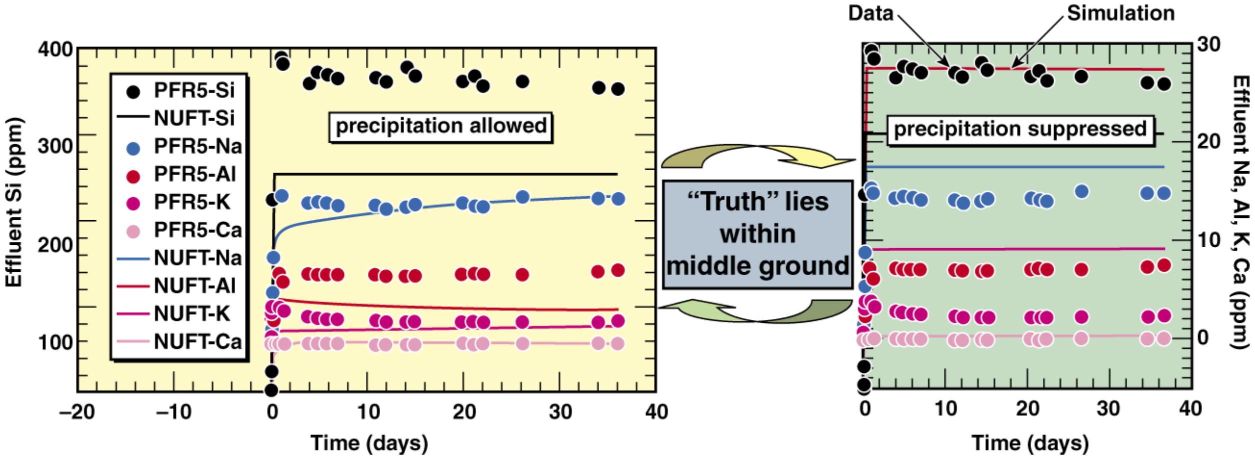


Figure 19. Plug-flow-reactor test of dissolution of tuff rock addresses the question of how small amounts of secondary mineral precipitation affect dissolution. Symbols show experimental data, lines indicate simulation. (Left) Precipitation allowed. (Right) All precipitation suppressed. Engine analysis will be used to find the correct degree and rate of mineral precipitation, among other results.

Application of the stochastic engine poses the following four fundamental challenges.

- Construction of the staged base representation. The relevant kinetic parameters for this experiment are readily grouped into two classes based on their relative uncertainty. The first-stage base representation includes the most uncertain kinetic data with all other parameters assigned fixed values. In the second stage, we include the remaining kinetic parameters. Potential subsequent stages will further include selected thermodynamic properties and process models, such as alternative representations of reactive surface areas.
- Determination of the prior probability distribution functions. The prior distribution of mineral properties will be constructed using upper and lower bounds and statistical methods, such as Parzen windows, as appropriate.
- Definition of the likelihood function. In order to determine the closeness of fit between predicted and observed effluent concentrations and dissolution/precipitation features, we will analyze experimental measurement errors and determine their statistical distribution.
- Evaluation of the posterior probability distribution. The posterior probability distribution for the mineral properties will be constructed from the sample frequencies generated by the engine. This distribution will be analyzed by selecting specific instances of the highest probability state space and graphically confirming their "closeness of fit" and by evaluating reasonableness (in terms of theoretical considerations) of these predicted high-probability parametric values and their associated uncertainties.

Successful application of the stochastic engine to the plug-flow reactor problem will lead to a new methodology for analysis of complex experiments. It will also demonstrate applicability of the engine to systems that have discrete parametric components. For the most complex problems, such as evaluating the performance of a nuclear waste repository, this kind of parametric analysis will allow us to break the problem up into more manageable segments. Once an accurate distribution of parameter values for a system such as the tuff dissolution case is available, it becomes part of the prior knowledge for large-scale experiments.

Engine Performance

MCMC Application: Development of New Stochastic Search Algorithms

The Metropolis algorithm (Metropolis et al., 1953) that is used by the engine in order to perform its stochastic search is guaranteed to eventually produce samples that obey the desired posterior probability distribution (Hastings, 1970). However, samples during an initial "burn-in" period must be discarded until the distribution of the generated samples converges to the desired distribution. After the burn-in period, there is some unspecified number of iterations necessary for accurately sampling the important regions of the state. The number of iterations needed for burn-in and for sampling the posterior depends on the particular problem and is usually determined by an appropriate suite of diagnostic tools.

Because of the high computational cost of running complex forward models, the number of necessary iterations should be made as small as possible. We have identified the following ways to help achieve this goal.

- Averaging of the state space to remove high-frequency features.
- Multi-level searching at different resolution scales.
- Adaptation of the search using information from previous iterations.

An Averaging Algorithm

One cause of slow convergence is the very steep local maxima in the likelihood function that is present for many inverse problems. We also discovered that systems with high-dimensionality, such as spatial problems, also suffer from a likelihood function that has a multitude of high-frequency features. Both aspects lead to slow convergence because the spatial step change (step size) of each search iteration must be very small to match these small-scale likelihood features. Many iterations will then be required in order to adequately sample the state space.

A solution to these difficulties is to perform some form of state space averaging over a resolution scale of interest. The posterior probability density $fX_{*}(x)$ is proportional to the product of the likelihood function $L\ (x)$ and the prior density $f_X\ (x)$,

$$f_{X*}(x) \propto L(x)\, f_X\, (x)$$

Simple averaging of both sides of this equation, however, will not preserve the form of this relationship because the average of a product is, in general, not the product of the averages. Even if this were true, it is not obvious, in general, how to generate samples from the averaged prior density. We have developed a new method, which we call transition neighborhood averaging, which gets around these problems (Nitao and Hanley, 2001a, 2001b).

Using this method of averaging, it can be shown that the averaged posterior density $\overline{f}_{X*}(x)$ is given by

$$\overline{f}_{X*}(x) \propto \overline{L}(x)\, f_X\, (x)$$

where $\overline{L}(x)$ is an averaged likelihood function. The averaging operator is equal to the expectation with respect to the transition probability of the proposal Markov chain. Note that the prior density function in the above expression is unchanged so that samples can be generated using the original prior distribution. The only modification to the basic MCMC algorithm is to replace the non-averaged likelihood function *L(x)* by $\overline{L}(x)$.

The averaged likelihood $\overline{L}(x)$ must be determined using sensitivity simulations or by an approximate method that uses displacements in the state at previous iterations. Another important feature of the new averaging method is that the scale of averaging is always the same as the magnitude of the step size.

Multi-Resolution Methods

The step size of the stochastic search algorithm must be comparable to the scale desired for resolving the posterior distribution. However, the smallness of this step size can reduce the efficient searching of the state space. A solution is to perform the search at different scales using multilevel stochastic search algorithms, such as simulated tempering (Marinari and Parisi, 1992) and tempered transitions (Neal, 1996). To apply these algorithms the likelihood function is smeared over coarser and coarser scales at higher and higher levels. The method of smearing that is usually presented in the literature is *ad hoc*, and no relationship is given for relating the step size and the amount of smearing.

The averaging method described in the preceding subsection is an ideal way to implement multiresolution algorithms by the use of larger and larger step sizes at coarser and coarser levels. The amount of smearing will then correspond exactly to the magnitude of the step size, and the step size will correspond to the desired resolution scale. In this way larger time steps at higher levels can rapidly traverse long distances in the state space while smaller steps at lower levels will be able to resolve the finer details of the posterior distribution.

Adaptive Markov Chain Monte Carlo Methods

Slow convergence, or “mixing” to the stationary posterior distribution is, in a large part, a consequence of the immense state space present in geological problems. Our solution to this problem is the development of adaptive methods whereby the algorithms learn from random variates that are generated for improving performance.

From a statistical perspective, the performance of an MCMC algorithm may be approached from two angles: rate of convergence to the stationary distribution and precision of estimated quantities along the sample path. The adaptive implementation of our MCMC routine learns from and optimally adapts through consideration of these two objectives with respect to the random variates generated by the algorithm. The ultimate goal is inexpensive adaptation, that is, optimization over these two objective functions at a minimal computational cost. To this end, we studied Gaussian approximations to posterior distributions of interest under which computation of convergence rates and precision measures, if not available in closed form, are readily available in computationally attractive formulations.

Recent developments are:

- We developed a general class of MCMC routines under which adaptive routines are most easily developed, analyzed, and implemented (Levine, 2001).
- We developed the underlying theory and constructed the adaptive Gibbs sampler algorithm (Levine and Casella, 2001).
- We analyzed the adaptive Gibbs sampler from a computational cost perspective (Levine et al., 2002a).
- We developed the underlying theory and constructed the adaptive Gibbs sampler algorithm (Levine and Casella, 2001).
- We extended the results from our adaptive Gibbs sampler to the more general Hastings sampler, within which the Metropolis sampler is a special case.

We plan to develop inexpensive implementations of the adaptive Hastings algorithm. This work is currently in preparation in Levine et al. (2002b).

Diagnostics and Posterior Inference

The stochastic engine is comprised of a variety of predictive forward models and specialized software modules rolled into a single integrated framework. Over the last year, this prototype has continued to evolve and grow in both complexity and functionality. The dynamic nature of this development demands that the validation of the embedded MCMC simulations be an ongoing priority. To address this need, run-time diagnostic tools capable of validating different aspects of the simulation have been developed, implemented and tested on both synthetic and real data. Several of these techniques are presented below. With a validated simulation, methodology for utilizing the generated posterior samples becomes a fundamental issue. In response, an effort to develop a statistical inference toolbox capable of supporting decision and risk analysis was initiated.

Simulation Diagnostics

The stochastic engine employs an MCMC algorithm to construct a probabilistic estimate of the state of nature that is consistent with observed data, modeling assumptions and prior knowledge. For our earth science application, the state of nature refers to a multi-attributed lithology map of a volume of earth. The engine produces a sample from the posterior distribution $f(x|data)$, which is the conditional probability distribution of the state of nature, given the data. This sample is a sequence of possible states of nature, $x^{(1)}$, $x^{(2)}$, …, $x^{(T)}$, … and is the entire basis for characterizing the posterior distribution and performing subsequent analysis. Theoretical results ensure that the sample eventually spans the entire posterior distribution and supports the estimation of the state frequencies that characterize the posterior. In mathematical jargon, the sample forms an ergodic Markov chain with stationary distribution *f*. This means that once the chain has taken a sufficient number of steps, T_0, the distribution of the state, $x^{(T)}$, at any step $T \geq T_0$ is exactly the posterior distribution, f. We call T_0 the "burn-in" time. Hence, the MCMC process begins at a particular state and after the burn-in period, it essentially forgets where it started. Determining the burn-in time, examining how effectively the sampling process is moving through the posterior distribution (called "mixing"), and validating known properties of the chain/distribution are all intended uses of the developed diagnostics.

Convergence to Burn-in

The convergence of the MCMC algorithm to burn-in is guaranteed when the proposal random walk is ergodic. This means that the samples produced will eventually be drawn from the posterior distribution. But, this result does not mention the actual rate of convergence. In fact, there are virtually no theoretical results on convergence rates which can be applied to most real world problems. Nevertheless, determining the burn-in point is critical to insure that any inference based upon the posterior distribution is not corrupted by "bad" samples. Hence, there is a strong need to develop a diagnostic method capable of assessing the convergence behavior of a given MCMC simulation.

The approach selected (due to Gelman and Rubin, 1992) employs multiple independent Markov chains to simultaneously estimate the burn-in period length T_0 and establish the claim of stationarity of the remaining samples. Each of the parallel chains has a different starting point, but they share a common limiting distribution, the posterior f. The Gelman-Rubin diagnostic detects when the variability between the chains settles down to a value that is expected when the chains are all sampling from a common distribution. To accomplish this, a parameter that is a multidimensional function of the state of nature must be identified for tracking throughout the simulation. For the SRS problem, a cross section of earth was split into two regions, upper and lower. A contiguous subregion is summarized by the triple $z = (z_1, z_2, z_3)$, where z_1 is the area, z_2 is the horizontal coordinate of the centroid, and z_3 is the vertical coordinate. By considering the largest contiguous subregion for each of the two cross section halves and two lithology types, say clay and silt, the dimensionality of the tracked parameter becomes p = 2x2x3 = 12.

The diagnostic tracks three quantities R^p, det$V$, and det$W$, which are functions of the p-dimensional states of the parallel chains for a moving and expanding window of steps. The window is characterized by a single parameter n. For example, n = 50 refers to the window of length 50 iterations from iteration 51 through iteration 100, and in general, the window of size n considers each chain within the iteration sequence n+1, n+2, …, 2n. The p-dimensional matrix W estimates the within chain variability for the window n, and the p-dimensional matrix B/n estimates the between chain variability for the same window. The pooled p-dimensional matrix

$$V = \left(\frac{n-1}{n}\right)W + \left(1 + \frac{1}{m}\right)\frac{B}{n}$$

where m is the number of chains, is an estimate of the covariance matrix of the posterior distribution of the parameter of interest. As n increases, i.e. the window moves and expands, the influence of the starting points on the individual chains diminishes, and the following conditions emerge:

- The within chain variation, summarized by det$W$, stabilizes. Typically, det$W$ increases, as new areas of modality of the parameter space are explored by the chains, before settling upon a limiting value.
- The pooled chain variation, summarized by det$V$, stabilizes, a result of the combined effect of the difference between chains, characterized by B/n, becoming negligible and the within chain variation stabilizing.
- The matrices V and W become close to one another. The measure of the distance between V and W is taken to be:

  $$R^p = \left(\frac{n-1}{n}\right) + \left(\frac{m+1}{m}\right)\lambda_1$$

  where λ_1 is the largest eigenvalue of the matrix $W^{-1}B/n$. As the distance between V and W diminishes, R^p approaches 1.

The diagnostic monitors R^p, det$V$, and det$W$, as a function of the window parameter n. For sufficiently large n, say $n \geq T_0$, the conditions, det$W$ and det$V$ approximately constant and R^p close to 1, are satisfied. The nearness of R^p to 1 suggests burn-in has occurred by step T_0, in that the between chain variation is negligible (hence the starting points have been forgotten); stabilization of the determinants in turn provides evidence that samples within the window, starting at iteration T_o+1, adequately characterize the stationary posterior distribution.

Example plots of the statistic R^p and the determinants det$V$ and det$W$ as functions of n are shown in Figures 20 and 21 for the Savannah River lithology problem, with dimension p = 8, based on analysis of the centroid but not the area. Four parallel chains were used in the simulation. The statistic R^8 approaches 1 and the determinants stabilize around $n = T_0 = 500$ iterations, the estimated burn-in length. Note that det$V$ always exceeds det$W$, and the two curves go up and down in tandem, ultimately converging.

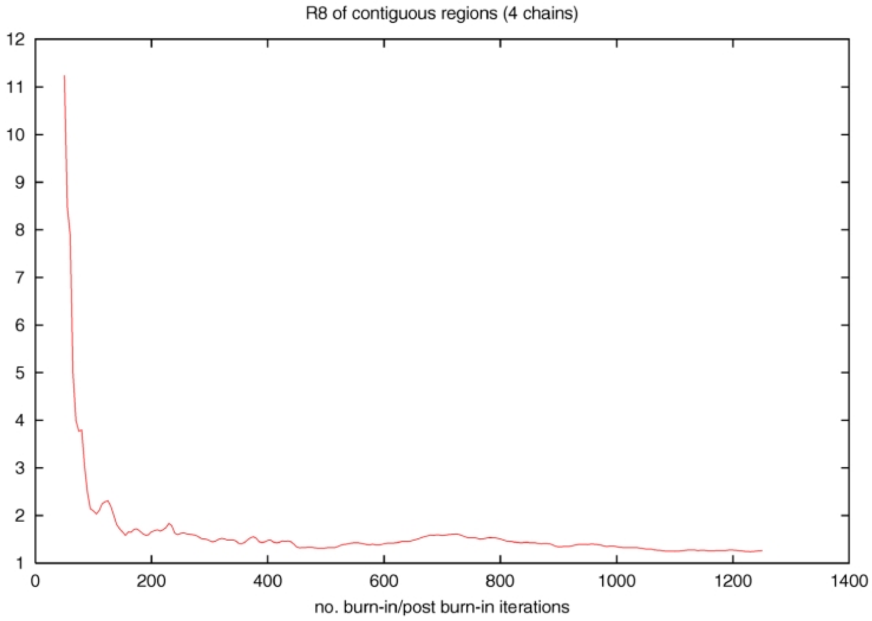


Figure 20. The *R^p* plot for Savannah River Problem, *p* = 8, *m* = 4. Note that it approaches 1 near *n* = 500.

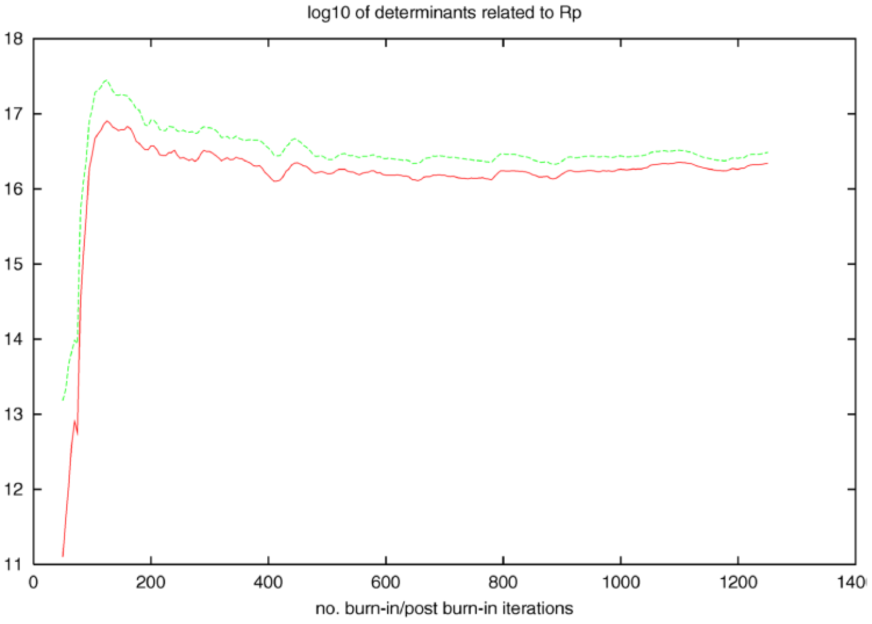


Figure 21. Plots of det*V* and det*W* for Savannah River problem, *p* = 8, *m* = 4. Note that both curves stabilize and approach one another near *n* = 500.

Mixing

The cumulated sum (or cusum) plot monitors, for a given MCMC process, the partial sums

$$S_t = \sum_{j=T_0+1}^{t} \left[h(x^{(j)}) - \overline{h(x)} \right] ,\ t = T_0+1,\ \ldots,\ n,$$

where $h(x)$ is a scalar parameter of interest, say the depth of the centroid of the largest contiguous region of a particular lithology type within a volume of earth, T_0 is the length of the burn-in period, and $\overline{h(x)}$ is the average value of $h(x)$ over the post burn-in steps $T_0+1, \ldots, n$. The plot displays S_t versus t for the range $t = T_0, T_0+1, \ldots, n$, with $S_{T_0} = 0$ and $S_n = 0$ by definition.

The cusum accumulates the differences between the value of a parameter at a given step and the overall average post burn-in value. It assesses the mixing speed of the chain (i.e., how fast a chain steps through the posterior distribution) and correlation between the $x^{(t)}$'s. If the chain is slowly mixing the values of $h(x^{(t)})$ do not change much in a neighborhood of t, and the plot is smoother and wanders farther from zero than if the chain is faster mixing, in which case the plot resembles Brownian motion.

The cusum is a subjective diagnostic that helps identify sampling schemes that are so slow mixing that alternative algorithms or parameterizations should be sought in order to more efficiently traverse the entire parameter space. Examples of the cusum are shown in Figures 22 and 23 for the dimension $p = 8$ Savannah River problem and for the dimension $p = 2$ Blob problem. In each case, the scalar parameter monitored is the depth of a contiguous region of specified lithology type, and the cusums of five parallel chains are plotted simultaneously. Figure 22 indicates faster mixing than Figure 23. Note that some chains are slower mixing than others, evidence of a chain's hanging around a particular mode for an extended period.

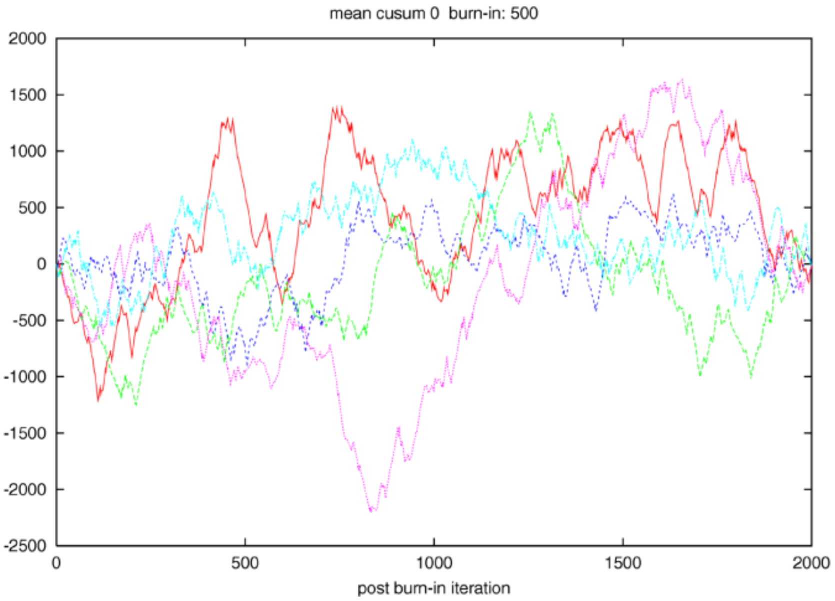


Figure 22. Cusum plots for 5 chains—Savannah River A/M Outfall lithology problem, *p* = 8.

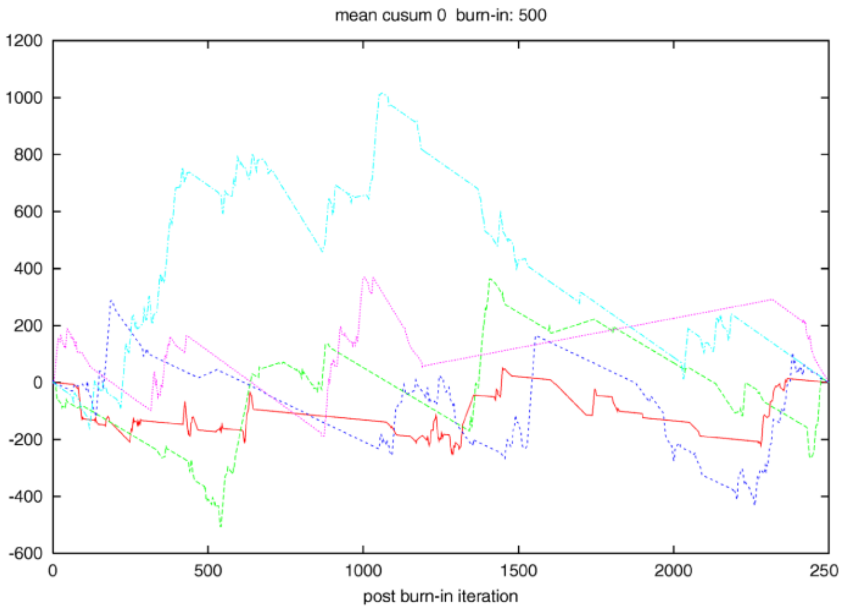


Figure 23. Cusum plots for 5 chains—blob problem, *p* = 2.

Distributional Characteristics

In addition to the above methods, two additional diagnostic tools have been developed, tested and incorporated into the current version of the stochastic engine. Both tools are concerned with validating known distributional properties of the generated sample. The first diagnostic focuses on the stationarity of the post burn-in portions of multiple chains. Specifically, the acceptance of a Kolmogorov-Smirnov two sample test performed on selected post burn-in subsamples provides a method for validating the existence of expected internal stationarity of the chain. The second diagnostic is concerned with testing the normality of a selected mean based upon post burn-in samples. In this case, a Kolmogorov-Smirnov one sample test provides the basis for confirming the normality of the mean. In both cases, a properly functioning simulation will exhibit characteristic behavior that approaches a known ideal, allowing for the identification of problems. Moreover, the absence of any pathological behavior is evidence of a properly functioning simulation. These tools combine with the prior two techniques to provide a well founded methodology for validating the engine's simulation process.

Posterior Inference Tools

The development of a statistical toolbox tailored to the type of information produced by the engine's MCMC simulation and supporting a variety of inference tasks is critical to maximizing its utility and application. Specifically, the engine generates a collection of samples from the posterior distribution defined on the possible states of nature. By construction, these samples embody the entirety of our available information and form the foundation for all subsequent posterior analyses. But, in their raw form, they are generally capable of only providing the basis for a coarse examination of the posterior distribution and its corresponding properties. To support moderate to detailed inference, including formal decision and risk analysis, specific methodologies require development and/or adaptation to the current problem domain. This effort has been initiated from two distinct perspectives. The first focuses on estimating and characterizing the marginal posterior distributions of lithology at each individual pixel; while, the second endeavor addresses the more challenging problem of estimating the joint posterior lithologic distribution. Both of these efforts are discussed in the following sections.

Marginal Distribution of Lithology

The Stochastic Engine uses new data to update existing information. The existing information is summarized by a prior distribution on the possible states of nature, while the updated version is summarized by the posterior distribution on these same states. Neither the prior nor the posterior is available in closed form, but the engine allows samples to be generated from both distributions. In its standard mode, the engine automatically generates samples from the posterior. But, if the engine is modified to accept all states proposed by the forward model sampler, the prior rather than the posterior is sampled. By running the engine in both modes, samples from each distribution can be generated and compared. This information allows the following quantities to be estimated.

- The uncertainty in our current understanding of the state of nature as indicated by the variability of the prior distribution.
- The uncertainty in our updated understanding of the state of nature as indicated by the variability of the posterior distribution.
- The effect of incorporating new data on our understanding of the state of nature as indicated by the change in variability present in the posterior and prior distributions.

In the discussion that follows, attention is restricted to estimating subsurface lithology for a two-dimensional cross section at the SRS. Prior information consists of well data that identifies lithology along a vertical borehole and spatial models embodied within the sampler TSIM. The new data consists of ERT data. We consider the problem of categorizing the lithology type (sand/gravel, silt, or clay) at each pixel in the cross section. Because of the restriction to individual pixels, the subsequent analysis focuses upon the estimation of the marginal prior and posterior distributions of lithology at each pixel separately. Approaches for dealing with the spatially contiguous joint estimation problem (lithologic distributions defined on entire images rather than single pixels) are much more difficult and are outlined in the next section.

For a given pixel, a sample of lithology classifications from the prior distribution can be modeled using a multivariate Bernoulli distribution, with parameters $(p_1^{(1)}, \ldots, p_{k+1}^{(1)})$ where k+1 is the number of lithology types (k+1 = 3 in the Savannah River example), and $p_j^{(1)}$ is the prior probability of categorizing the pixel as having lithology type j, with $p_1^{(1)} + \ldots + p_{k+1}^{(1)} = 1$. Similarly, a posterior sample of lithology types can be modeled by a multivariate Bernoulli distribution, with parameters $(p_1^{(2)}, \ldots, p_{k+1}^{(2)})$, where $p_j^{(2)}$ is the posterior probability of categorizing the pixel as having lithology type j, with $p_1^{(2)} + \ldots + p_{k+1}^{(2)} = 1$. Since the prior and posterior distributions are of the same type, we will simplify notation wherever possible by eliminating the superscript, with the understanding that the development applies to the prior and posterior alike.

The multivariate Bernoulli random variable will be denoted as $\mathbf{X} = (X_1, \ldots, X_k)^T$, where X_j = 1 if the pixel is categorized as having lithology type *j* and X_j = 0 otherwise, with $X_1 + \ldots + X_{k+1} = 1$. Sufficient information about the probability parameters is contained in the sampled frequencies of the various categories, N_j, j = 1, …, k+1, $N_1 + \ldots + N_{k+1} = N$, where *N* is the total sample size, and $N_j = X_{j1} + \ldots + X_{jN}$ with X_{ji} being the indicator of classification as lithology type *j* at sample *i*. For example, in the Savannah River case, $N_3^{(2)}$ represents the total number of times in the posterior sample the pixel was categorized as type "clay".

The inherent degree of uncertainty or variability in classifying the lithology type at the given pixel is a function of the $\{p_j\}$. We consider a scalar measure of uncertainty, called the *generalized variance,* that is based on the $k \times k$ dispersion matrix of the X_{ji} For this model, the generalized variance equals the determinant of the dispersion matrix, and is simply the product

$$D = \prod_{j=1}^{k+1} p_j .$$

The larger the value of *D*, the greater is our uncertainty. Smaller values imply less uncertainty. If the lithology type is known with certainty, say of type *t*, then p_t = 1, and p_j = 0 for $j \neq t$, and *D* = 0. When the lithology type is not known, the value of p_j is estimated by the relative frequency from the sample, $\hat{p}_j = N_j / N$. Such estimates of the p_j result in an estimate $\hat{D}$ of the generalized variance.

The estimated generalized variance $\hat{D}$ provides a measure of the degree of uncertainty in assessing the state of nature. By calculating $\hat{D}$ at each pixel we can compare the relative amounts of uncertainty and produce a coarse ordering. In Figure 24 we see graphs of the generalized variance for SRS runs of sample size 2500 for the prior and 2400 for the posterior. These sample size values represent the number of post burn-in iterations. The metric $\hat{D}$ is linearized to a 0 to 1 scale and the contour plots are colored so that darker colors indicate smaller variability: the color scheme is dark brown->brown->green->yellow->pale-yellow as the metric ranges from 0 to 1. Note that there exist a number of band-like subregions of comparable variability.

At any given pixel the data would be expected to have some influence on the prior lithology classification probabilities { $p_j^{(1)}$} so that the posterior probabilities { $p_j^{(2)}$} are different. Hopefully, this difference would be in the direction of reducing uncertainty, but this is not guaranteed. (For example, the prior could be overly compact and new data shows this to be an unreasonable assumption, and attempts to improve the characterization by forcing the posterior to be more dispersed.) To examine the influence of new data, the following statistical tests were implemented and applied.

- A χ^2 test of *equality* of two multinomial distributions. This procedure examines whether there is any significant difference (in either direction) between the prior and posterior pixel classification probabilities.
- A test for a *reduction* in variability as measured by the generalized variance metric.

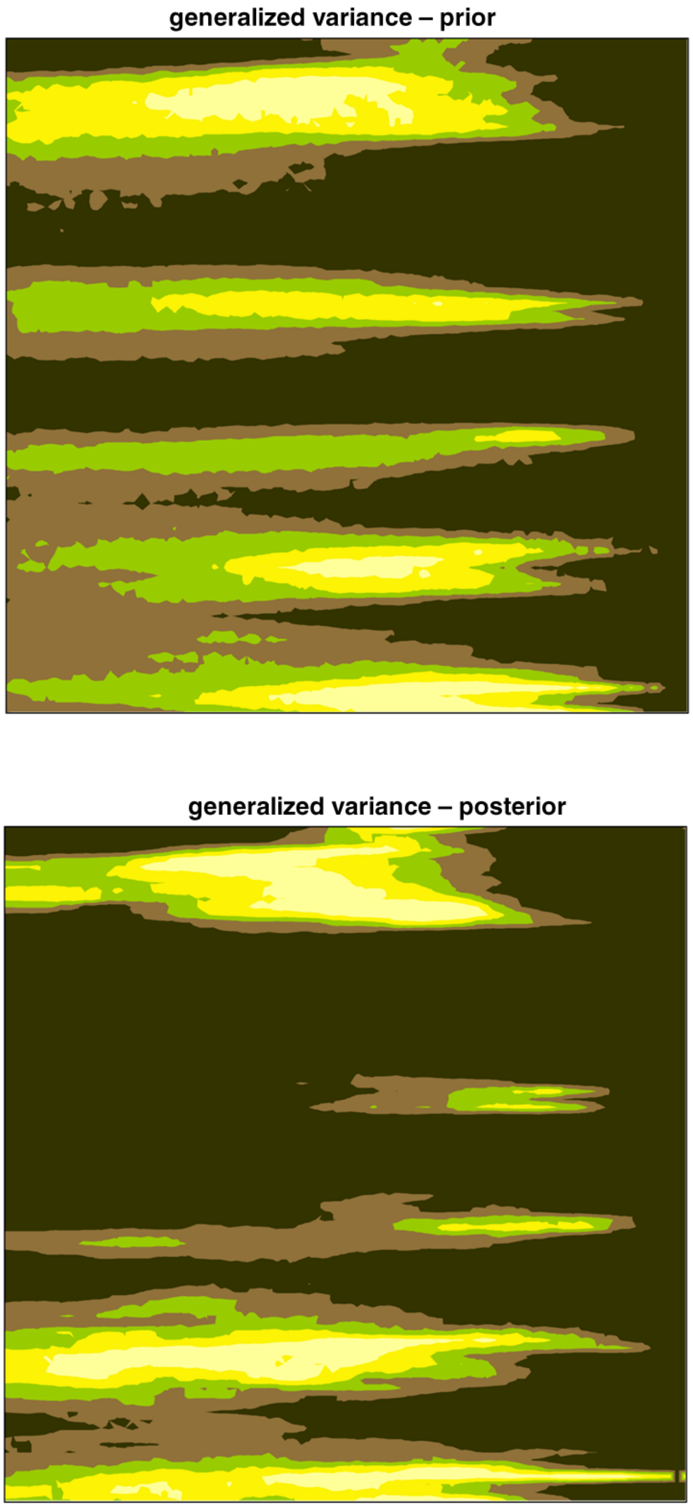


Figure 24. (Top) The estimated prior generalized variance for the Savannah River site example, based on a post-burn-in sample of size 2500 from the prior distribution. Darker colors indicate smaller amounts of pixel-level uncertainty. (Bottom) The estimated posterior generalized variance, based on a post burn-in sample of size 2400 from the posterior distribution.

Results of the χ^2 test on the previous Savannah River example are shown in Figure 25. Illustrated are the p-values at each pixel. A p-value is the probability that under the assumption of equal distributions (i.e., the prior and posterior are the same for the pixel in question), a χ^2 random variable with *k* degrees of freedom would exceed the statistic,

$$\chi^2_{obs} = \sum_{i=1}^{2} \sum_{j=1}^{k+1} \frac{\left[N_j^{(i)} - \frac{N^{(i)}}{N^{(1)}+N^{(2)}}\left(N_j^{(1)} + N_j^{(2)}\right)\right]^2}{\frac{N^{(i)}}{N^{(1)}+N^{(2)}}\left(N_j^{(1)} + N_j^{(2)}\right)}$$

calculated from the generated samples. A small value, such as 0.05, is evidence that the observed frequencies are very unlikely if the distributions are equal and hence the new data changed the classification probabilities. Note that Figure 25 displays an abundance of low p-values, and hence the data had a significant influence at most pixels.

The test for the reduction in variability is based upon large sample statistics which ensure, under the assumption of equal distributions, that the statistic,

$$Z = \sqrt{\frac{N^{(2)}}{2k}}\left(\frac{\hat{D}^{(2)}}{\hat{D}^{(1)}} - 1\right),$$

has an approximate standard normal distribution. Evidence of departure from the equality hypothesis in the direction of reduced variability is provided if the posterior estimated generalized variance is sufficiently smaller than the prior version to make *Z* significantly small.

Instead of using a formal test to contrast changes in distributional variability, one may simply compare the generalized variances of the posterior and prior distributions via an examination of their log ratio. Figure 26 displays these ratios for the Savannah River Site example. Observe that for several horizontal bands of pixels there appears to be a significant reduction in variability from the prior to the posterior, as indicated by the brown colors which correspond to small ratios of the posterior generalized variance to the prior generalized variance. In these cases, the effect of the new data is a significant reduction in our degree of uncertainty. On the other hand, the blue colored areas indicate a significant increase in our degree of uncertainty. This is because, as we observed in Figure 25, the new data affect nearly every pixel significantly, one way or the other, relative to our prior assessment.

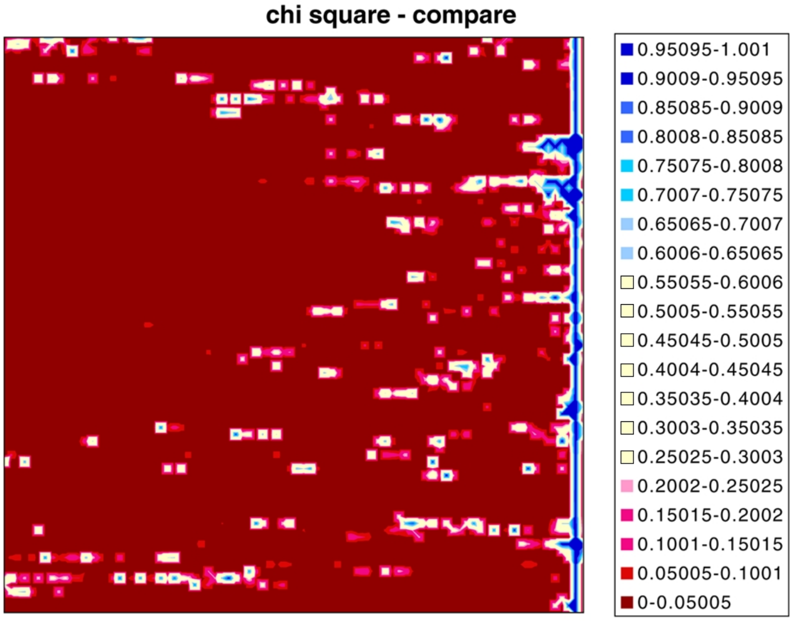


Figure 25. χ^2 p-values for testing the equality of the prior and posterior classification probabilities in the Savannah River site example, based on a post-burn-in sample of sizes 2500 and 2400 from the prior and posterior distributions, respectively. Small p-values are evidence of significant differences in the prior and posterior probabilities.

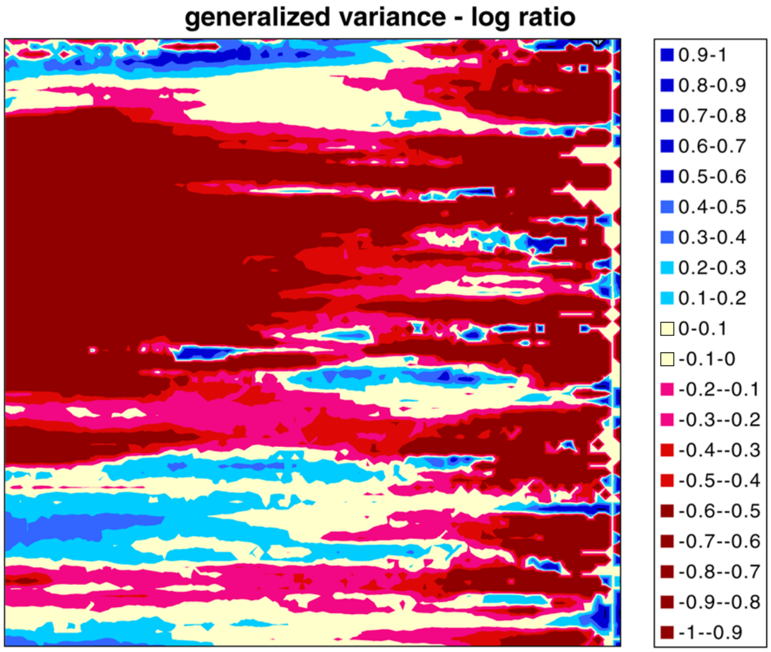


Figure 26. Log ratios of the posterior generalized variance to the prior generalized variance for the classification probabilities in the Savannah River Site example, based upon a post-burn-in sample of sizes 2500 and 2400 from the prior and posterior distributions. Smaller log ratio values (indicated by the dark brown shades) indicate larger reductions in uncertainty when ERT data are combined with the prior distribution to produce the posterior distribution.

Finally, if one were required to reach a conclusion on what the lithology is at a particular pixel, a reasonable choice would be the lithology type with the highest posterior classification probability. Specifically, one would classify the pixel as lithology type *m*, where

$$\hat{p}_m^{(2)} = \max\{\hat{p}_j^{(2)} : 1 \leq j \leq k+1\}$$

and $\hat{p}_m^{(2)} = N_m^{(2)} / N^{(2)}$ is the modal posterior classification probability for the given pixel. From large sample statistics, a lower confidence bound for $p_m^{(2)}$ is

$$\underline{p}_m^{(2)} = \hat{p}_m^{(2)} - z\left(\hat{p}_m^{(2)}(1 - \hat{p}_m^{(2)}) / N^{(2)}\right)^{1/2},$$

where z denotes the normal quantile for the desired confidence level. For example, one uses z = 1.28 for 90% confidence. Figure 27 depicts 90% lower confidence bounds for each pixel's modal posterior classification probability. Hence, if a lower bound at a given pixel is $\underline{p}_m^{(2)} = 0.95$, then one asserts with 90% confidence that the posterior classification probability for lithology type *m* is at least 0.95. It is interesting to compare the results given in Figure 27 to similar lower bounds based on samples from the prior distribution (not shown). The logarithm of the ratio of posterior versus prior modal probability lower bounds is displayed in Figure 28. The darker areas (positive values) indicate higher modal probabilities for the posterior, the lighter areas (negative values) indicate higher modal probabilities for the prior, and the white areas (values close to zero) indicate the posterior and prior modal probabilities are about the same. Hence, in the dark areas, the combination of data and the prior yielded higher modal probabilities and more certainty, while the other areas became less certain (yellow shades) or were unchanged (gray or white shades).

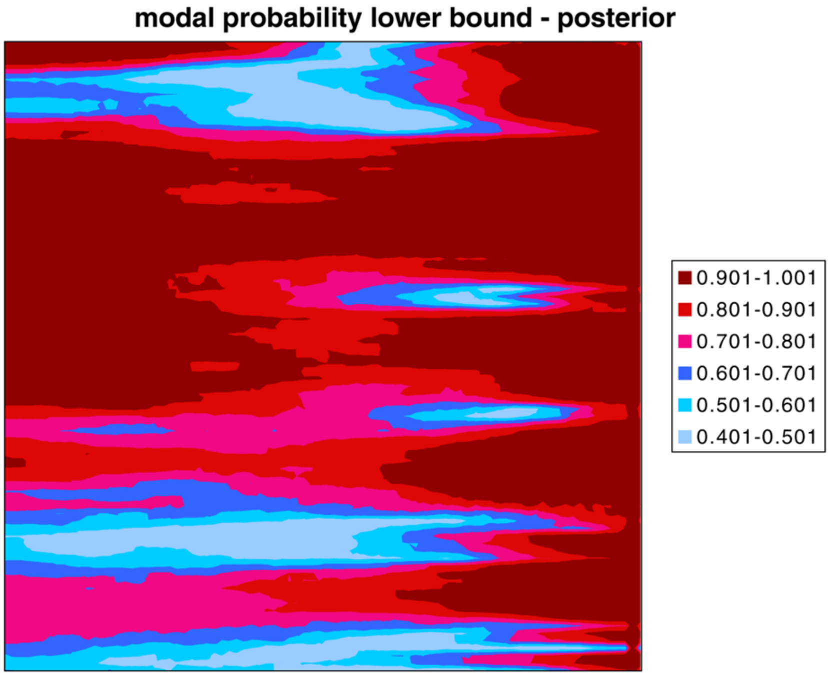


Figure 27. 90% lower confidence bounds on the modal posterior classification probability in the Savannah River site example, based on a post-burn-in sample of size 2400 from the posterior distribution.

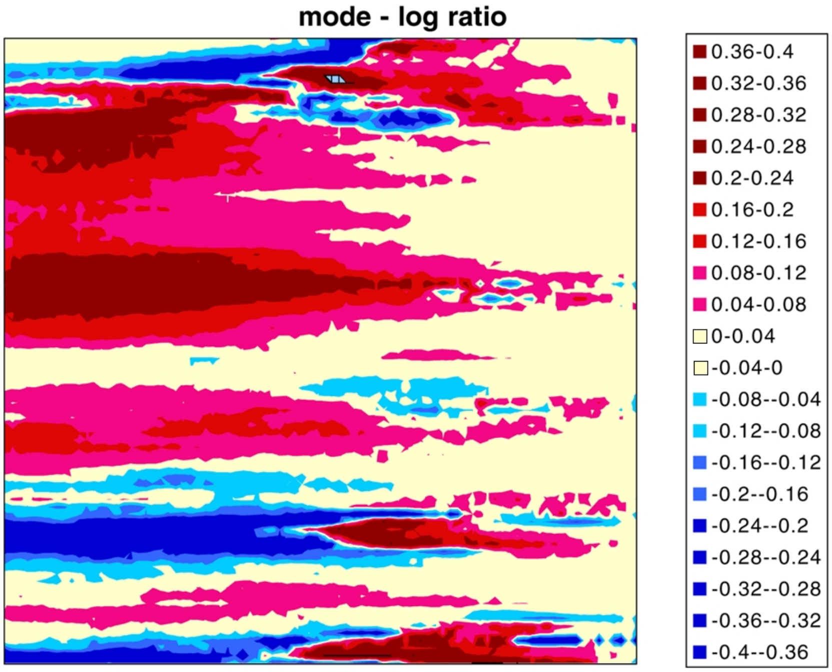


Figure 28. The logarithm of the ratio of posterior versus prior modal classification probability lower bounds in the Savannah River site example, where the post-burn-in prior and posterior sample sizes were 2500 and 2400, respectively.

Pixel level inference methods provide useful insight into the marginal behavior of the lithologic distribution. In fact, this marginal information can often indicate coarse behavior and characteristics of the joint distribution of lithology. But, due to the loss of spatial information that occurs when pixels are characterized individually, this extension of marginal results to the joint case must be taken with a grain of salt. To bridge the gap, research into characterizing the joint lithologic distribution is under way and is the topic of the next section.

Joint Distribution of Lithology

A lithologic image typically has hundreds, often thousands of pixels in it. Posterior samples can run into the thousands in number, making meaningful inference computationally challenging. This situation is compounded by the collection of samples being very sparse in their distributional support. Successful inference methodologies must avoid these dimensionality problems, but at the same time be computationally efficient. One technique which addresses these issues is global clustering, that is grouping together images that are similar in a well-defined sense to one another. The cluster centers and the associated cluster frequencies offer a good guide to the distribution under study. Recently developed clustering algorithms (PROCLUS, CLIQUE), which address problems similar to ours, have demonstrated promising results. Another approach which complements a variety of inference methodologies (including clustering) involves the transformation of the sampled images into a lower dimensional space. Commonly used transformation methods like Karhunen-Loeve Expansion, Fourier Transform, and Singular Value Decomposition all turn out to be inappropriate for our subsurface application. But, a particular type of wavelet transformation appears to hold promise as an inference preprocessing step. Specifically, a uniform wavelet shrinkage method significantly reduces the dimension of the wavelet transforms of the lithology images, making clustering and other inference algorithms easier to apply. These approaches are discussed briefly in the subsequent sections.

To better understand the posterior distribution, one can apply a form of clustering to the generated posterior samples. Specifically, clustering will yield a relatively small number of clusters for which within-cluster similarities are much greater than those between distinct clusters. This structure will enable us to provide a probability estimate for each cluster with a physical representation of the cluster center in terms of lithologic characteristics. But, to leverage clustering methodology in the subsurface application domain, any selected algorithm must be adapted to handle labeled data. Specifically, clustering algorithms rely upon the definition of a similarity measure which is applied to pairs of samples. However, since our representation provides pixel values which are labels indicating a pixel's lithologic type, the similarity measure will require modification to handle nominal data. For example, one must provide a suitable definition of the cluster center and a procedure to update it when a new sample is added or deleted when required by the algorithm. To this end, the several concepts have to be adapted to the nominal case and formalized.

To define a similarity measure capable of handling labeled data, a lithologic image is viewed as a vector of n binary images—one image corresponding to each of the *n* possible pixel labels. In each binary image, a pixel has a value of 1 (or 0) to indicate the presence (or absence) of the lithology type in the original image. For example, in the case of three lithology types (e.g., clay, gravel, silt) one can write $I = (I_c, I_g, I_s)$. Then, if a second lithology image *J* has the representation $J = (J_c, J_g, J_s)$, the similarity of the image pair, denoted by *S(I,J*) can be computed as follows.

$$S(I,J) = \sum_{l=1}^{n} CC_0(I_l, J_l)$$

where $CC_0(I_l, J_l)$ denotes the cross-correlation at lag(0,0) between binary images I_l and J_l. Note that this similarity measure can be generalized by defining it as the average of itself and eight lagged cross-correlation's, where the lags are based on an 8-neighborhood of each pixel (for 2D images, 26-neighborhood). Additionally, the sum in the above definition can be replaced by a weighted sum, where the weights form a normalized *n*-tuple embodying the relative importance of the specific lithology types for the given application.

When considering the assignment of a new sample to a cluster, the similarity of that sample to the cluster center and a corresponding user-specified threshold *T* are used to determine membership. An individual cluster is parameterized by the number of samples used to form that cluster and its *n* label constituents. For each pixel, the constituents are normalized over the *n* labels yielding a sum equal to 1. For example, suppose there are three lithologies and consider a particular pixel with center components (c, g, s) = (.6, .1, .3) in a cluster of n=10 samples. The addition of a new sample (1, 0, 0) (i.e. a clay pixel) results in the cluster parameters being updated to n =11 and the pixel's center components become (7, 1, 3) / 11. In other words, the pixel in the updated cluster consists of 7 clay, 1 gravel and 3 silt pixels. Observe that with similarity well defined, the threshold $T \in [0,1]$ effectively determines the number of clusters formed varying from 1 to *N* = number of samples. The choice of T is obviously application dependent – often proceeding by trial and error.

Suppose there are three lithologies and the algorithm produced *K* clusters $C_1, C_2 \ldots C_K$ arranged in decreasing order of their sizes $N_1, N_2, \ldots N_K$. with associated cluster centers given by the triples,

$$\left(I_{c1}, I_{g1}, I_{s1}\right), \left(I_{c2}, I_{g2}, I_{s2}\right), \ldots, \left(I_{cK}, I_{gK}, I_{sK}\right)$$

For simplicity, one may label the cluster centers by the dominant lithology in each pixel. For example, if the largest component in the triple corresponds to clay and it exceeds a threshold of say .5, then it would be labeled a clay pixel. If the dominant lithology of the cluster center does not exceed the chosen threshold, then a neutral lithology can be introduced to represent that indeterminate pixel. The process is repeated for each pixel in the image, yielding a single image for each cluster center which can be regarded as representative of its constituent members. Hence, we can produce a frequency distribution of the clusters which are represented as single lithologic images. This allows one to assign probabilities to states ranging from the most likely to the least likely lithologic explanations of nature (i.e., cluster centers). The effectiveness of this method when applied to the current problem domain is presently unknown, but, in similar spatial applications, clustering methodology has provided an effective method for characterizing high-dimensional distribution behavior.

The development of an effective approach to simplifying the characterization of high-dimensional lithologic distributions is under way. This effort is focused on the development of a computationally efficient and near information preserving transformation of lithologic images into a lower dimensional space via a "uniform" wavelet shrinkage method. The uniform manner of this approach distinguishes it from the usual wavelet compression methods. Specifically, in a typical wavelet shrinkage, coefficients are truncated based on their magnitudes. The retained coefficients, potentially can be located in different regions of the 2-D wavelet coefficient space for different lithology samples. So for any clustering procedure to be successful one must use the full dimension of the wavelet coefficient space, namely the dimension of the lithology image space. This renders the transformation useless for our purposes. This problem can be circumvented if the significant coefficients (coefficients that produce a good approximation to the original images when the inverse wavelet transformation is applied) are localized uniformly in a small region of the coefficient space.

Empirical study confirms that wavelet shrinkage implemented by restricting the coefficients of a Daubechies-4 wavelet transform in a narrow L-shaped region with the corner at the origin in the wavelet coefficient space of the lithology images, retains the broad lithologic structure present in the posterior samples. Although this empirical observation appears to be fairly robust for lithology images, the underlying mathematical reason for this is not entirely clear and is currently under investigation. Possibly it is because of the somewhat layered structure of the lithology distribution. Additionally, the information loss is dependent on the choice of the width of the L-shaped region. A global measure is provided by the cross-correlation between the compressed and the original lithology image. With the compression factor varying from 8 to 32, the mean cross-correlation drops from 0.94 to 0.82. Since an information preserving compression would yield a cross-correlation of 1.0, some information is lost. But, visual observation of selected test cases do not indicate a significant degradation relative to our application domain. The illustrations provided in Figure 29 are based on a random sample of 500 selected from a set of ~10,000 posterior samples of lithology images based on the Savannah River site. The figure displays the wavelet compressed images along with the original images of 10 randomly chosen lithology images from the posterior samples. From the bottom up each row represents five original images and the row immediately above shows the corresponding compressed images. Note that the visual degradation is minor. The wavelet transform is Daubechies with coefficient 4 implemented by the IDL software. The compression factor is 8. Figure 30 displays histograms of the three cross-correlation coefficients at lag (0, 0) between the original and the compressed image pairs based on the 500 samples and three levels of compression. The figures in the panel viewed from top to bottom correspond to compression factors: 8, 16 and 32. As the compression factor increases, the mean correlation decreases from 0.94 to 0.82.

Understanding a posterior distribution generally involves some form of density estimation. In high dimension this is difficult because of the relatively small number of samples available. The problem is compounded further because of concentration of samples in subspaces. One possible approach to this problem is based upon clustering and using the cluster center and frequency like a high-dimensional histogram approximation of the posterior distribution. Since the cluster centers actually represent a lithologic distribution, they provide a good idea of typical lithologies present in the posterior support and how likely they are to occur. To enhance the effectiveness of our density estimation, a successful effort has been made to reduce the dimension of the data using uniform wavelet shrinkage.

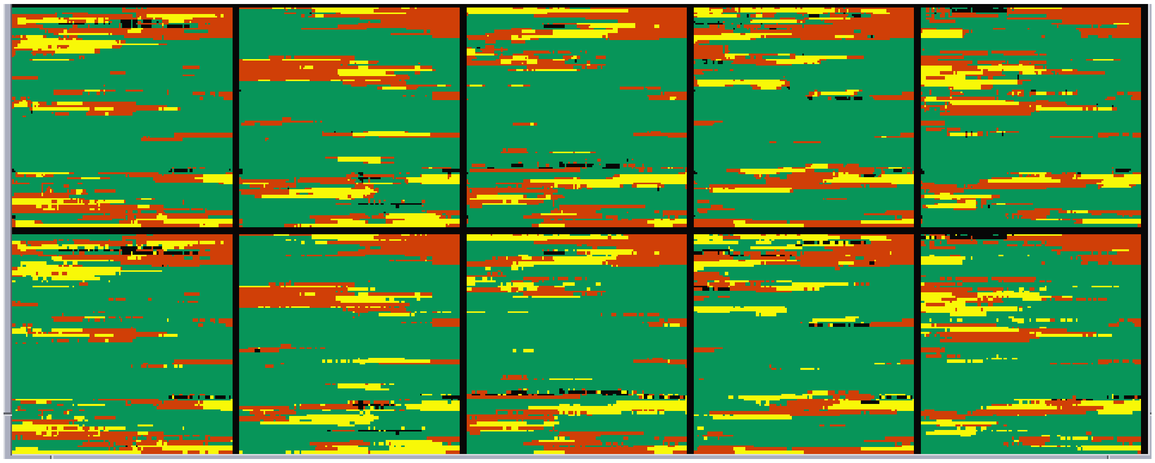


Figure 29. Wavelet compressed images corresponding to five lithologic states generated from a Stochastic Engine simulation based on Savannah River data. (Bottom row) Five original images. (Top row) The corresponding compressed images. The wavelet transform is Daubechies with coefficient 4 implemented by the IDL software. The compression factor is 8.

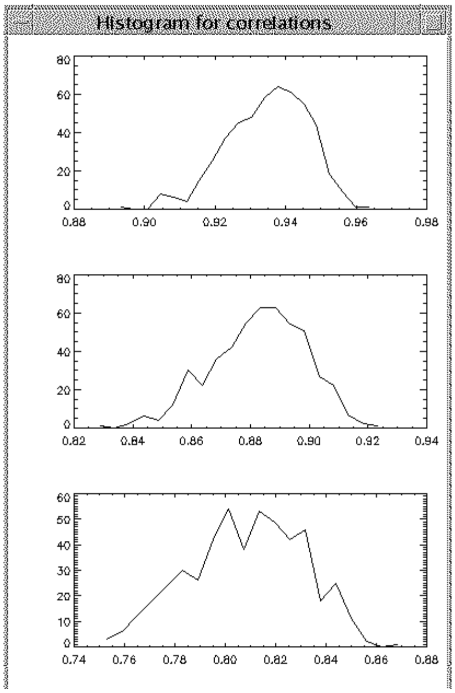


Figure 30. Histograms of the three cross-correlation coefficients at lag (0, 0) between the original and the compressed image pairs based on the 500 samples and three levels of compression. From top to bottom, the histograms correspond to compression factors of 8, 16 and 32. As the factor increases, the mean correlation decreases from 0.94 to 0.82.

Needed Performance: Computational Costs and Task Parallelization

For complex spatial problems the main computational cost in the stochastic search algorithm will be in the time required to solve the forward models. As the number of computational cells increases, especially as the problem domain is increased from 2D to 3D (Figure 31), the computational and memory limits of a single workstation can easily be reached. Thus, we must devise ways to subdivide up the problems into smaller tasks that run on separate CPU's. There are several levels of task parallelization possible.

At the coarsest level of task granularity, we take advantage of the fact that the multiple MCMC chains that are necessary for computing convergence diagnostics and ensuring adequate coverage of the state space are chains that are statistically and functionally independent. The only communication involved is the transfer of accepted states from each chain to a central collection point where it can be post-processed. For a 1 million cell problem the amount of information that must be transferred is on the order of 1 MB for each state. The time required to send this information over a LAN is small compared to the time it takes to run the forward model and this communication can be done asynchronously from the computation. This form of parallelization has been implemented into our engine for local workstations. In the future, this capability will be expanded to run jobs on other machines, such as the Teracluster. Note that this form of parallelization works independently of the type of forward model that is used.

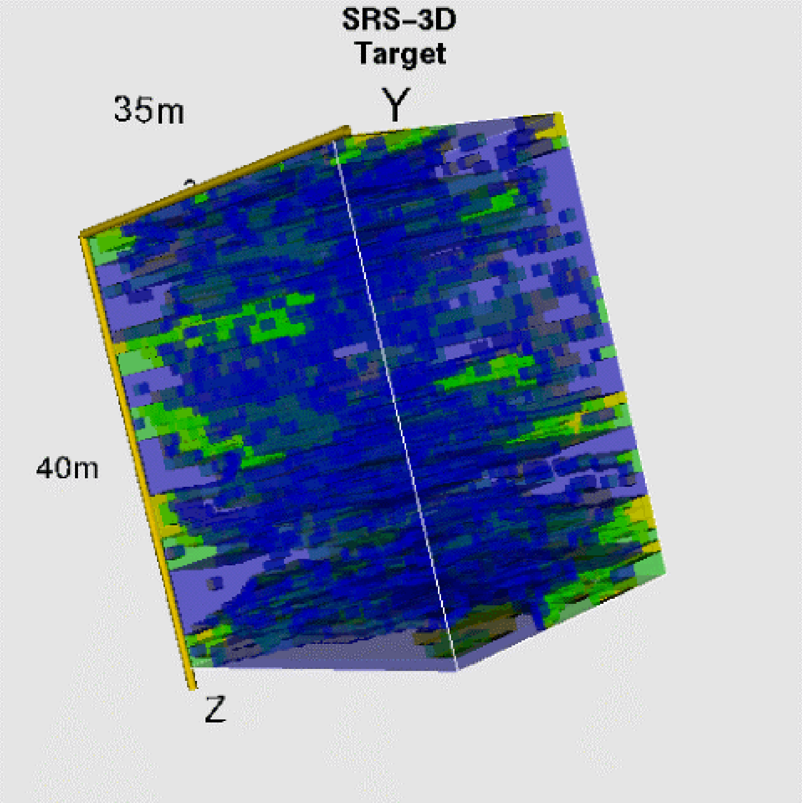


Figure 31. All engine components now operate in three dimensions.

At the next lower level of granularity, the forward model computation for each independent chain can also be parallelized. For 2D electrical resistive tomography there are 10 independent linear solves required (the ERT code is actually pseudo-3D; the 10 solves correspond to 10 spectral components to model the 3rd dimension). For the 3D ERT forward model, the number of independent linear solves is equal to the number of borehole electrodes. Each linear solve (or a fixed number of linear solves) can, therefore, be done on its own workstation. A prototype for this method of parallelization has recently been implemented on a local workstation cluster. Performance enhancements to this system and porting to the Teracluster will take place during the coming year. A single linear solve of a 100×100×100 cell ERT problem is estimated to take approximately 30 seconds on a 1GHz workstation, using around 600 MB of physical memory. This size of problem appears to be a rough upper limit on the largest ERT problem that is feasible with this approach. The two linear solves needed by a flow and transport model would be done on a separate processor.

Larger problems will require solving each linear system on multiple processors such as multiple workstations on their own fast network or on the Teracluster 2000 machine. Thus, each chain runs on its own independent group of processors, and each group is subdivided into nearly independent subgroups with each subgroup performing a single linear solve (see Figure 32). This stage of parallelization will be implemented during the coming year.

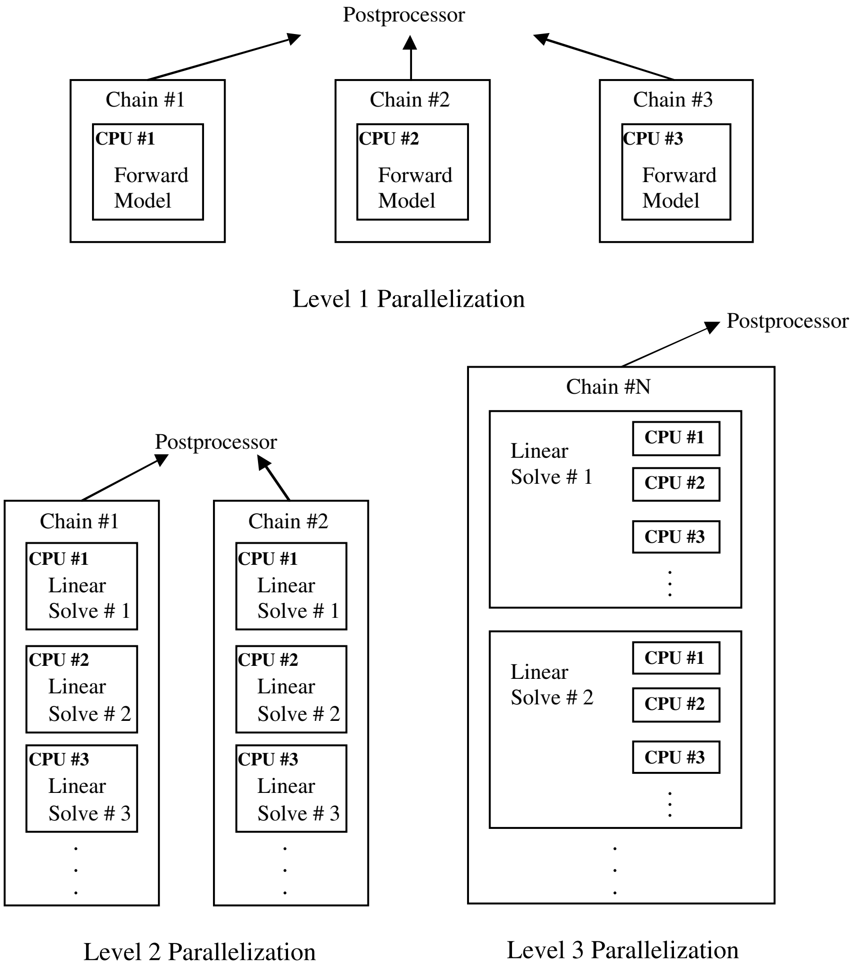


Figure 32. Strategies for parallelization of the engine software.

References

Carle, S.F.; Fogg, G.E. (1997), Modeling spatial variability with one and multidimensional continuous-lag Markov chains, *Mathematical Geology* **29**(7), 891-918

Carle, S.F. (1997), Implementation schemes for avoiding artifact discontinuities in simulated annealing, *Mathematical Geology* **29**(2), 231-244.

Carle, S.F., Labolle, E.M., Weissmann, G.S., Van Brocklin, D, Fogg, G.E. (1998), Geostatistical simulation of hydrofacies architecture, a transition probability/Markov approach: in Fraser, GS; Davis, JM, *Hydrogeologic Models of Sedimentary Aquifers, Concepts in Hydrogeology and Environmental Geology* No. 1, SEPM (Society for Sedimentary Geology) Special Publication, pp. 147-170.

Carle, S. F. (1996), A Transition Probability-Based Approach to Geostatistical Characterization of Hydrostratigraphic Architecture. Ph.D. dissertation, Department of Land, Air and Water resources, University of California, Davis.

Deutsch, C. V., and Journel, A. G. (1992), *Geostatistical Software Library and User's Guide*, Oxford University Press, New York.

Gelman, A., and Rubin, D. B. (1992), Inference from iterative simulation using multiple sequences*, Statistical Science* **7**, 457-511.

Hastings, W.K. (1970), Monte Carlo sampling methods using Markov chains and their applications, *Biometrika* **57**, 97-109.

Johnson, J.W., Knauss, K.G., Glassley, W.E., DeLoach, L.D., and Tompson, A.F.B. (1998), Reactive transport modeling of plug-flow reactor experiments: quartz and tuff dissolution at 240°C, *J. Hydrology* **209**, 81-111.

Johnson, J.W., Nitao, J.J., Steefel, C.I, and Knauss, K.G. (2001), Reactive transport modeling of geologic CO_2 sequestration in saline aquifers: the influence of intra-aquifer shales and the relative effectiveness of structural, solubility, and mineral trapping during prograde and retrograde sequestration, *Proceedings of the First National Conference on Carbon Sequestration*, Washington, DC, May 14-17, 2001, 60 p.; Lawrence Livermore National Laboratory, Livermore, CA, UCRL-JC-146932, www.netl.doe.gov/publications/proceedings/01/carbon_seq/P28.pdf.

Johnson, J.W., Nitao, J.J., Tompson, A.F.B., Steefel, C.I., et al. (1999), 21st-century tools for modeling reactive transport in dynamic geologic systems of economic and environmental significance, *Earth and Environmental Sciences 1999 Annual Report*, p. 7-11, Lawrence Livermore National Laboratory, Livermore, CA, UCRL-LR-126434-98.

Johnson, J.W., Oelkers, E.H., and Helgeson, H.C. (1992), SUPCRT92: A software package for calculating the standard molal thermodynamic properties of minerals, gases, aqueous species, and reactions from 1 to 5000 bars and 0 to 1000C, *Computers and Geosciences* **18**(7), 899-947.

Levine, R. A. (2001), *A Note on Markov Chain Monte Carlo Sweep Strategies*, Lawrence Livermore National Laboratory, Livermore, CA, UCRL-JC-145678; submitted to *Statistics and Computing*.

Levine, R. A. and Casella, G. (2001), *Optimizing Random Scan Gibbs Samplers*, Lawrence Livermore National Laboratory, Livermore, CA, UCRL-JC-145679; submitted to the *Annals of Statistics*.

Levine, R. A., Yu, Z., Hanley, W. G., Nitao, J. J. (2002a), *Implementing Random Scan Gibbs Samplers, in preparation*.

Levine, R. A., Yu, Z., Hanley, W. G, Nitao, J. J. (2002b), *Implementing Componentwise Hastings Samplers*, in preparation.

Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., and Teller, E. (1953), Equation of state calculations by fast computing machines, *J. Chemical Physics* **21**(6).

Mosegaard, K. and Tarantola, A. (1995), Monte Carlo sampling of solutions to inverse problems, *J. Geosphys. Res*. **100**(B7), 12,431-12,447.

Neal, R.M. (1996), Sampling from multimodal distributions using tempered transitions, *Statistics and Computing* **6**, 353-366.

Newmark, R.L. et al. (2002), Stochastic Engine: Direct Incorporation of Measurements Into Predictive Simulations, *Proceedings of the International Groundwater Symposium "Bridging the Gap Between Measurement and Modeling in Heterogeneous Media*, Berkeley, CA, March 25–28, 2002; Lawrence Livermore National Laboratory, Livermore, CA, UCRL-JC-145116.

Nitao, J.J. (1998a), *Reference Manual for the NUFT Flow and Transport Code, Version 2.0*, Lawrence Livermore National Laboratory, Livermore, CA, UCRL-MA-130651.

Nitao., J.J. (1998b), *User's Manual for the USNT Module of the NUFT code, Version 2.0*, Lawrence Livermore National Laboratory, Livermore, CA, UCRL-MA-130653.

Nitao, J.J., and Hanley, W.G. (2001a), *Use of Averaged Densities to Promote Mixing of the Metropolis MCMC Algorithm*, Lawrence Livermore National Laboratory, Livermore, CA, UCRL-ID 145681-DR.

Nitao, J.J., and Hanley, W.G. (2001b), *Metropolis-Type Chains with Finite Memory and an Averaging Algorithm to Promote Mixing*, Lawrence Livermore National Laboratory, Livermore, CA, UCRL-ID 145680-DR.

University of California
Lawrence Livermore National Laboratory
Technical Information Department
Livermore, CA 94551